DESIGN OF NONLINEAR SECOND-ORDER DYNAMIC SYSTEMS TO MEET ARBITRARY PERFORMANCE CRITERIA

By

JAMES EDWIN BOSE

Bachelor of Science
Oklahoma State University
Stillwater, Oklahoma
1960

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1961

Submitted to the faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the degree of
DOCTOR OF PHILOSOPHY
July, 1966

DESIGN OF NONLINEAR SECOND-ORDER DYNAMIC

SYSTEMS TO MEET ARBITRARY

PERFORMANCE CRITERIA

Thesis Approved:

E C Fitch

Thesis Adviser

P. Doanfeld

D Leroy Folks

K. N. Reed

J H Boggs

Dean of the Graduate College

ACKNOWLEDGMENTS

My sincere appreciation is extended to the following individuals and organizations:

Dr. E. C. Fitch, who served as my major adviser and graduate committee chairman. His interest, encouragement, and guidance were invaluable during my graduate study.

My graduate committee, composed of Professors P. Dransfield, J. L. Folks, and K. N. Reid, who rendered guidance and counsel throughout my graduate program.

The Caterpillar Tractor Company, Joliet, Illinois, and the School of Mechanical Engineering, Oklahoma State University, for their financial support.

Messrs. James O. Matous, Jerry Mrazek, and K. K. Gowdy for their friendship and many hours of constructive discussion.

The project members I have worked with at the Mechanical Engineering Laboratory.

Mr. M. A. Ross for his valuable assistance in the preparation of the manuscript.

My father, Edward J. Bose, for his encouragement and financial aid throughout my education.

My wife, Arnola, for her many hours spent in typing the manuscript and for her continued encouragement during my graduate study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

a_{i1}	. .	constant coefficient of the control parameter u_{i1}, to be determined in the analysis method
a_n	. .	constant coefficient of general equation
$a_n(x, t)$	. .	coefficient of general equation
B	. .	damping factor
C_8	. .	acceleration constant
C_0	. .	corrected velocity
f_i	. .	a function of state of the fixed system
F_d	. .	damping force
F_x	. .	force in the x-direction
F_i	. .	a function of the control system which defines the i-th state of the controlled system
F_{od}	. .	orifice damping force
F_{sp}	. .	spring force
$F(t)$	. .	system input force
$(F_x)s$	. .	steady flow force in the x-direction
$(F_x)us$	. .	unsteady flow force in the x-direction
g_i	. .	function of the control parameters u_{ij} which defines the control of the system
h	. .	function of the system variables
K	. .	spring constant
K_1	. .	rise time correction factor
K_2	. .	time to overshoot and settling time correction factor

K_{ij} . . constant coefficient of the fixed system U_{ij}

M . . system mass

Mv . . momentum of a mass particle

OS . . overshoot

P_i . . number of parameters of the fixed system U_{ij}

q_i . . number of parameters of the control vector u_{il}

ρ . . density

T_d . . delay time

T_{OS} . . time to maximum value

T_{RT} . . rise time or time to first zero error

T_{ST} . . settling time

U_{ij} . . a parameter of the fixed control system

u_{il} . . a control parameter which is a function of the state of the system

X_i . . the i-th of a set of dependent variables which describe the state or the system, a state variable

X . . displacement or response of the system

$\dot{X}$ . . velocity of the system

$\ddot{X}$ . . acceleration of the system

$\ddot{X}(0)$ . . acceleration at time $t = 0$

ω_n . . natural undamped frequency

ζ . . damping ratio

CHAPTER I

INTRODUCTION

General Introduction

In synthesis, the designer must integrate components into a functional system which will operate in a specified manner. To do this, he must:

(1) establish a set of system requirements, and

(2) synthesize a system to meet these requirements.

The basic premise of this process is that proper and efficient design must be firmly based on a physical understanding of system and component characteristics. This physical understanding is obtained by extensive use of mathematical models of the system components. The mathematical models are obtained by appropriate application of the physical laws (laws of conservation, momentum, energy, etc.) governing the behavior of system components.

The behavior of any particular system is partially governed by operational conditions imposed upon it. A number of the system requirements are defined as "unalterable" in that they must be met and cannot be varied or changed to bring about a desired system performance. Other system requirements are somewhat adjustable and are designated as "alterable," subject to change in subsequent design steps. A mathematical model represents a physical system in much the same way as the laboratory model simulates it. The mathematical models of engineering systems, which, in equation form, describe the set of alterable (variable) and unalterable (fixed) system dynamic characteristics, consists commonly of nonlinear differential equations with time varying coefficients of the form

$$a_n(X, t) \frac{d^n X}{dt^n} + a_{n-1}(X, t) \frac{d^{n-1} X}{dt^{n-1}} + \ldots + a_0(X, t) = F(t). \tag{1-1}$$

No general methods of analysis or synthesis, such as those developed for linear systems, have been developed for real physical systems described by Equation (1-1). A few writers have developed direct methods for treating some nonlinearities, but these methods are generally limited to specific cases and are not easily extended to solve more general situations.

The compensation technique for a system described by Equation (1-1) is a trial-and-error process in which characteristics are assumed for certain alterable elements, the combination analyzed, changes made, another analysis performed, etc., until satisfactory performance has been achieved. The trial-and-error process involved is essentially a series of "experiments" performed on paper, using mathematical models. The determination or fitting of these alterable coefficients to obtain a specific desired output comes from a study of the basic dynamic equations of the physical system. If the coefficients can be changed or fitted such that a set of desired performance characteristics are obtained, the synthesis of the system is achieved.

In many cases the desired performance of the system is not attainable from a manipulation of the variable coefficients of the system because of design constraints or requirements placed on it. These changes may take the form of:

(1) additional equipment selection, and/or

(2) the utilization of system nonlinearities.

These additions require utilizing elements with essentially fixed characteristics in order to achieve a desired system performance.

For a linear system of the form

$$a_n \frac{d^n X}{dt^n} + a_{n-1} \frac{d^{n-1} X}{dt^{n-1}} + \cdots + a_1 X + a_0 = F(t) \qquad (1\text{-}2)$$

where a_n, a_{n-1}, $\ldots$, a_1, a_0 are real numbers and $a_n \neq 0$, synthesis by the introduction of additional compensatory type of equipment is well documented (21) (8). However, addition of equipment will result in an increase in system order and complexity. Linear theory does not cover the utilization of system nonlinearities to bring about the required control.

While no general closed form analytical methods are available to allow the designer to take advantage of the inherent nonlinearities that exist or that can be designed into the system, solutions using the analog and digital computers are commonplace.

The advantages of using system nonlinearities are twofold:

(1) A nonlinear control system can be made to have a better response than a linear control system of the same order.

(2) The amount of system hardware is likely to be less.

These factors can be illustrated by an example.

Example

Consider the spring mass system shown in Figure 1-1. Its mathematical model or dynamic equation is

$$M\ddot{X} + B\dot{X} + KX = F(t). \tag{1-3}$$

Equation (1-3) is a linear differential equation, as mass M, damping factor B, and spring constant K are all constant.

The response of this system to a unit step input in terms of two parameters ω_n and ζ is shown in Figure 1-1. The type of response depends on the system parameters M, K, and B. Very often, a compromise between the overdamped and the underdamped system would be advantageous. For the underdamped system, the rise time is sufficiently short but the overshoot is excessive. Increasing the damping coefficient B will decrease the overshoot; however, the rise time is increased, thus producing a "sluggish" system. A compromise such as that indicated by the broken line in Figure 1-1 would produce the desirable characteristics of each of the two linear responses.

One way to achieve this compromise would be by the introduction of a suitable nonlinear dynamic characteristic into the system. The nonlinear characteristic may be inherent in the system or may be designed into the system by using a component having the required characteristic. For the system of Figure 1-1, a nonlinear characteristic may be introduced by modifying the system so that its equation is

$$M\ddot{X} + B(x)\dot{X} + K(x)X = F(t) \tag{1-4}$$

The coefficients B(x) and K(x) are now functions of the dependent

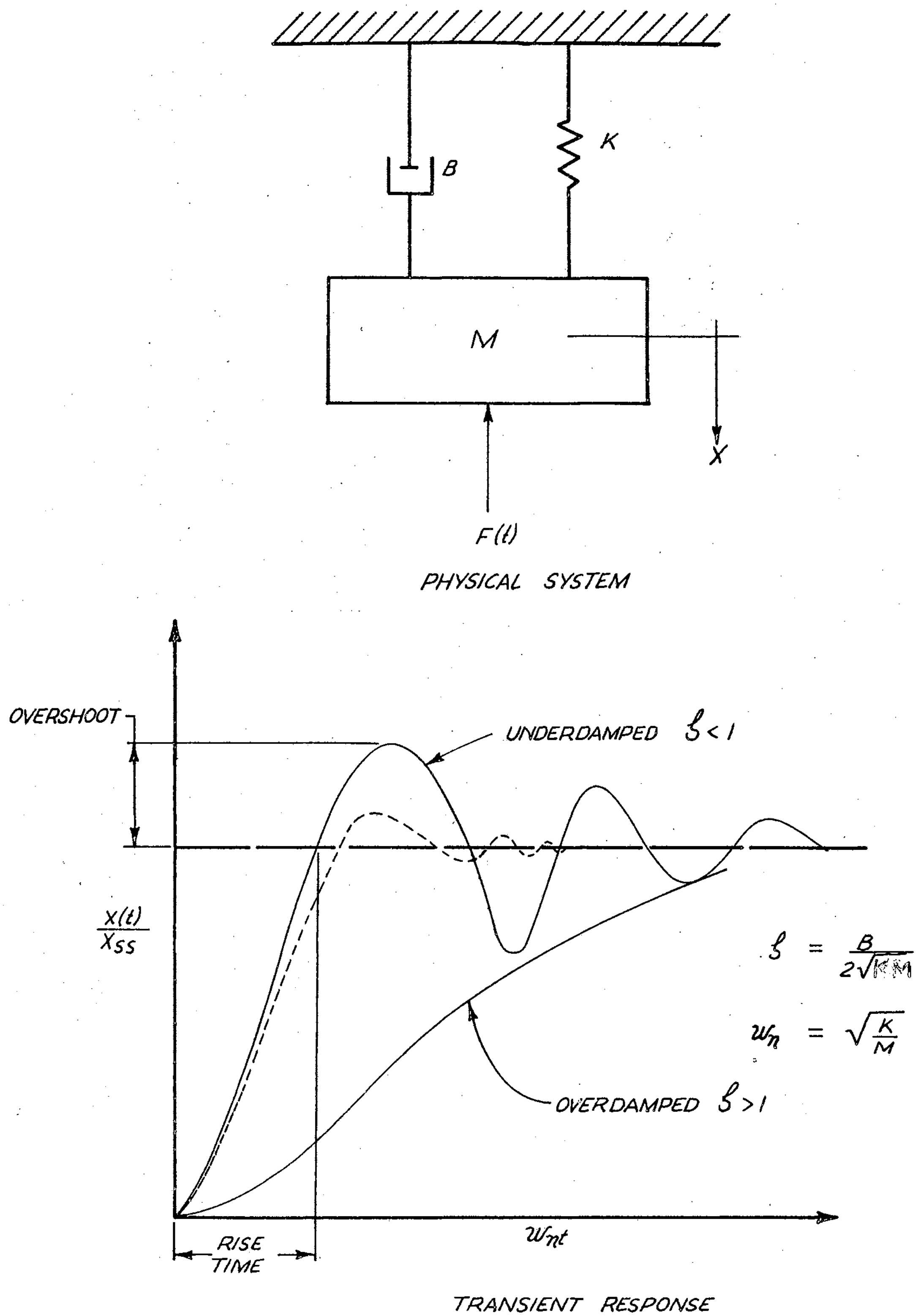


Figure 1-1. Spring Mass System and Its
Dynamic Characteristics

variables x, thus making the system nonlinear. If the coefficients can be assigned such that a set of desired performance characteristics are met, then a system synthesis method is achieved. The form of B(x) and K(x) will depend on the particular type of physical components used. For example, if a sharp edge orifice is used in the piston of the damper of Figure 1-2, Equation (1-4) takes the form

$$M\ddot{X} + \dot{X}|\dot{X}| + KX = F(t). \tag{1-5}$$

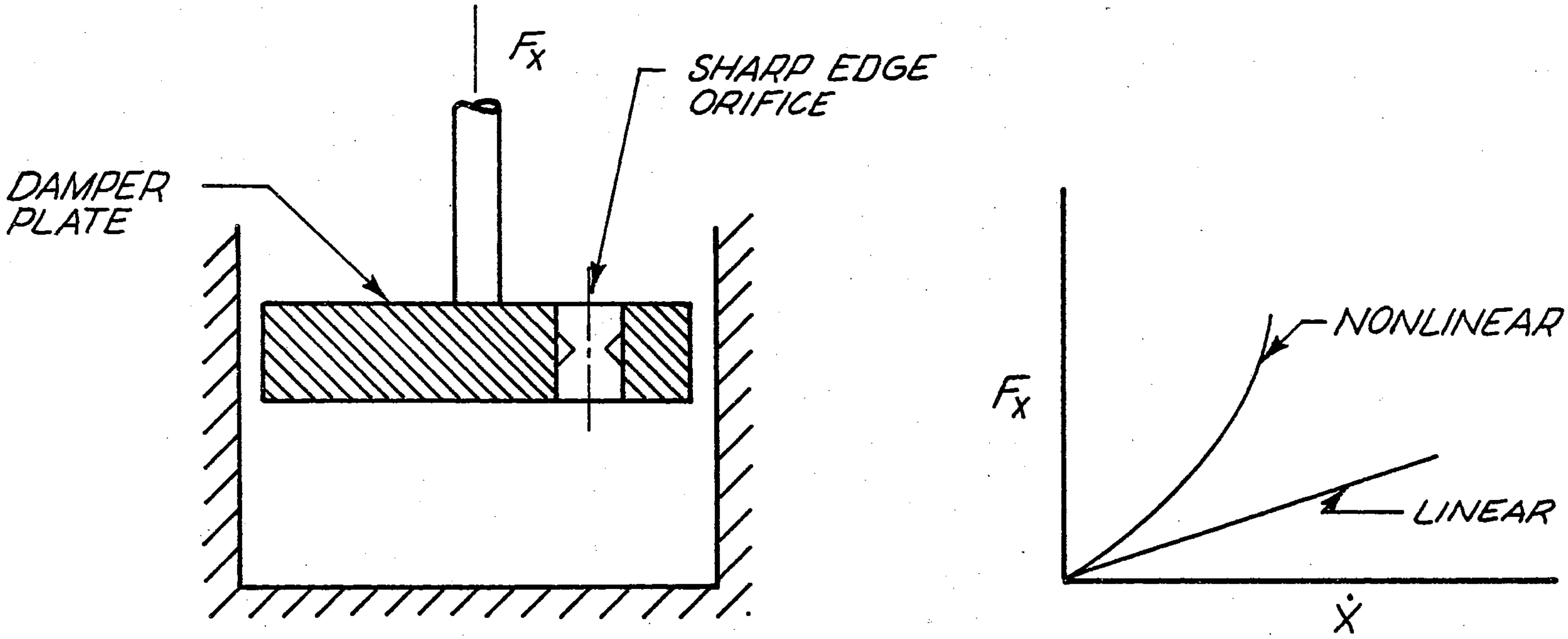
Placing the modified damper into the system can result in a more desirable response characteristic (the broken line of Figure 1-1) without an addition of extra equipment into the system.

The acceptability of this type of modification of the damper depends on how well the damping factor B(x) is able to achieve a desired system response. An example of achieving a desired response by the introduction of system nonlinearities is presented in Chapter III.

Statement of the Problem

The purpose of the study is to develop a procedure for forcing a dynamic phsyical system to operate in a desired optimum manner. The type of system considered in this thesis is governed by an ordinary, second order, stationary differential equation. Briefly, if a particular system behaves in an undesirable manner, what system parameter can be adjusted or added to enable it to behave in the desired manner? Specifically, how may achievable nonlinear characteristics be used to obtain the desired response characteristics?

Mathematically, the problem may be stated as follows: Given a dynamic control system which is represented by a set of ordinary



Nonlinear Damper

Damper Characteristics

Figure 1-2. Nonlinear Damper and Its Characteristics differential equations of the form

$$\frac{dX_i}{dt} = f_i(X_1, X_2, \ldots, X_N) \tag{1-6}$$

what set of coefficients of a predetermined functional form control vector g_i can be found such that the system

$$\frac{dX_i}{dt} = f_i(X_1, X_2, \ldots, X_N) + g_i(X_1, X_2, \ldots, X_N) \tag{1-7}$$

performs in a specified manner?

The forcing of a system to perform in a desired manner is a problem of determining the coefficients of alterable system elements which will enable the system to exhibit a desired performance.

The control vector g_i defined in Equation (1-7) will be limited to those terms which are inherent to the system or whcih can be designed into the existing system.

CHAPTER II

LITERATURE SURVEY

The analysis and compensation of feedback control systems containing nonlinearities, whether inherent or deliberately put into the system, has recently become a subject of wide interest. A general solution of nonlinear equations has not been found; consequently, the methods of nonlinear analysis are valid for only certain types of nonlinear problems. The transient response of a nonlinear system to a specific input function can be found by one of the following methods:

(1) performing an actual test on the system with a given input.

(2) representing the physical system on an analog or digital computer and performing the tests on the computer, or

(3) using a step-by-step graphical procedure to calculate the response.

These methods are strictly trial-and-error solutions if the nonlinear system is to be designed according to some predescribed transient response characteristics.

The problems associated with compensating physical systems are still somewhat unsolved. For linear systems, the analysis techniques have been extensively developed. Some of the techniques which have received considerable acceptance are the root-locus method and the polar diagram method (21). Other analytical approaches which involve minimizing error integrals have been explored but so far have been limited to linear systems (18). Because of the lack of general solutions to nonlinear problems, a large number of specific analysis techniques have been developed.

One widely accepted technique for nonlinear analysis is the phase-plane method, which gives an insight to the synthesis problem by presenting a perspective of all transient possibilities. Unfortunately, the consideration of adding a compensation network is usually rejected in this approach because the phase-plane method cannot be simply extended to systems possessing many energy-storage elements.

The describing-function method is useful in determining the stability of nonlinear systems but cannot be directly applied in the optimization of the system design. This method, based on quasi-linearization, converts the nonlinear elements to an equivalent linear model whose parameters depend on the amplitude and frequency of a sinusoidal input. This method is particularly valuable because of its great similarity to the well-known frequency-response method for designing linear systems. Attempts to extend the describing-function method to study the nonlinear transient problem have not appeared promising. Correlation between the frequency response and the transient response of a nonlinear system is unlikely in view of the invalidity of the principle of superposition in nonlinear analysis. The describing-function method is normally classified as a frequency-response method rather than a time-domain technique and is based on an analysis that neglects the effect of harmonics in the system. Gibson (9) presents an analytical approach by means of an inverse describing function. However, the method is restricted to a single nonlinear element or multiple nonlinearities which can be represented by a single nonlinearity.

Chen (7) developed a method of studying the transient response of a large class of nonlinear control systems to a step input. The basis of his method is a quasi-linearization in which the nonlinear system is converted to an equivalent linear model. An evaluation of the transient characteristics of the quasi-linearized system through linear servo techniques furnishes an insight to the problem of designing the nonlinear control system to meet a predescribed transient response. Chen's method differs from the describing-function method in that a transient type of test function instead of a sinusoidal test function is used to obtain the quasi-linear model. Solutions using the method by Chen can be obtained only after going through a large number of steps of approximations, equivalent representations, etc. Even after a complete representation is obtained, several additional steps are required to translate the results in the time domain and to obtain actual approximate transient responses. Although the proposed method is straightforward when the nonlinearity considered is simple, the complexity in applying the method increases rapidly when there are several nonlinear elements in the system.

Potts, Ornstein, and Clymer (19) used a steepest-descent method on an analog computer to establish the parameters in a mathematical model simulation of a human operator. The required derivatives were obtained on an analog computer for an essentially error-free output.

Bellman, Kagiwada, and Kalaba (3) proposed a method of quasi-linearization for identification purposes; and Kumar and Shridhar (13) applied this method to stationary plants. The methods developed by

Bellman, Kagiwada, and Kalaba should not be overlooked as a possible procedure for analyzing system performance.

Bernhart (4) presented a numerical method whereby an ordinary non-linear constant coefficient differential equation was fitted to a set of input-output variables of an unknown system. His method produced very good results. The fundamental problem was to determine what set of operations are present such that the system may be characterized by an ordinary differential equation of the form

$$W[X(t),\ Y(t)]_i = 0 \quad , \quad i = 1,\ 2,\ \ldots,\ N \tag{2-1}$$

where W represents a sum of differential and multiplicative operations on the variables $X(t)$ and $Y(t)$ and N is the number of distinct evaluations of each of the operations. For example, Equation (2-1) can be characterized by an ordinary differential equation of the form

$$A_1\ddot{Y} + A_2\dot{Y}Y + A_3\dot{X}Y = X. \tag{2-2}$$

With Bernhart's method, the system parameters or constant coefficients A_1, A_2, and A_3 in Equation (2-2) may be determined with the associated implication that should any of the parameters vanish the corresponding operation is a non-contributory relation between the input and output. The fitting of Equation (2-2) to the system data was accomplished by defining a weighted residual R_i associated with each of the N discrete equations (Equation 2-1).

$$R_i = P[A_k,\ X(t),\ Y(t)]_i(W_i) \tag{2-3}$$

where W_i denotes the relative weight of the residual R_i and k is the number of operations included in the system equation. The sum of squares of the weighted residuals

$$G(A_k) = \Sigma R_i^2 \tag{2-4}$$

is minimized by taking the partial derivative of G with respect to A_k and setting the results equal to zero

$$\frac{\partial G}{\partial A_k} = 0 \quad , \quad k = 1, 2, \ldots, 1. \tag{2-5}$$

Hove (12) extended Bernhart's work by developing a numerical method for synthesizing the response of a dynamic process. The dynamic process to be synthesized was described by a system of ordinary differential equations of the form

$$\frac{dX_i}{dt} = f_i(X_1, X_2, \ldots, X_n, t) + g_i(u_{i1}, u_{i2}, \ldots, u_{im_i}) \tag{2-6}$$

where $X_1, X_2, \ldots, X_n$ are the state variables, i.e., these variables describe the state of the system at each value of the independent variable time t. The relation f_i defines the fixed process and

$$g_i = a_{i1}u_{i1} + a_{i2}u_{i2} + \ldots + a_{im_i}u_{im_i} \tag{2-7}$$

defines the control parameters which are used to bring about a desired response. Equation (2-6) is then arranged as

$$a_{i1}u_{i1} + \ldots + a_{im_i}u_{im_i} = \frac{dX_i}{dt} - f_i \tag{2-8}$$

where the right side is the fixed part of the model and the left side is variable in the selection of the parameters $a_{im_i}u_{im_i}$. The synthesis procedure is based on considering m states, $m >> \sup(m_i)$, which are derived from specifications. $\operatorname{Sup}(m_i)$ is the maximum number of terms of the control vector m. The resulting set of overdetermined equations is then solved for the unknown coefficients by the least squares approach (14) (11). This requires choosing the coefficients a_{ij}, $j = 1, 2, \ldots, m_i$ so that the sum of the squares of the difference between the fixed and variable parts of Equation (2-8), at each of the m states, is a minimum with respect to the a_{ij}. That is

$$[(a_{ij}u_{ij} + \cdots + a_{im_i}u_{im_i} + f_i - \frac{dX_i}{dt})^2]_i = [(\text{Residual})^2]_i$$

is minimized with respect to the a_{ij}. This minimization process is developed in detail in Chapter III.

The results of Hove's method are most encouraging from the standpoint of fitting a system to a desired response. Figures 2-1, 2-2, and 2-3 are the curve fits accomplished by Hove (12) on pages 24, 25, and 26 of his thesis. The variable X_1 is the desired response, X_2 is the first derivative or velocity, and $\dot{X}_2$ is the acceleration of the system. The desired or specified response curves were constructed by Hove (12) in an arbitrary manner.

The control vector g_i used in the synthesis procedure was obtained by considering all combinations of X_1 and X_2 of second degree and above, up to but not including terms of fifth degree. The difficulties in fitting this specific response with a nonlinear differential equation are twofold:

(1) The desired response is defined completely for all values of time. No latitude is allowed in a choice of state variable representation.

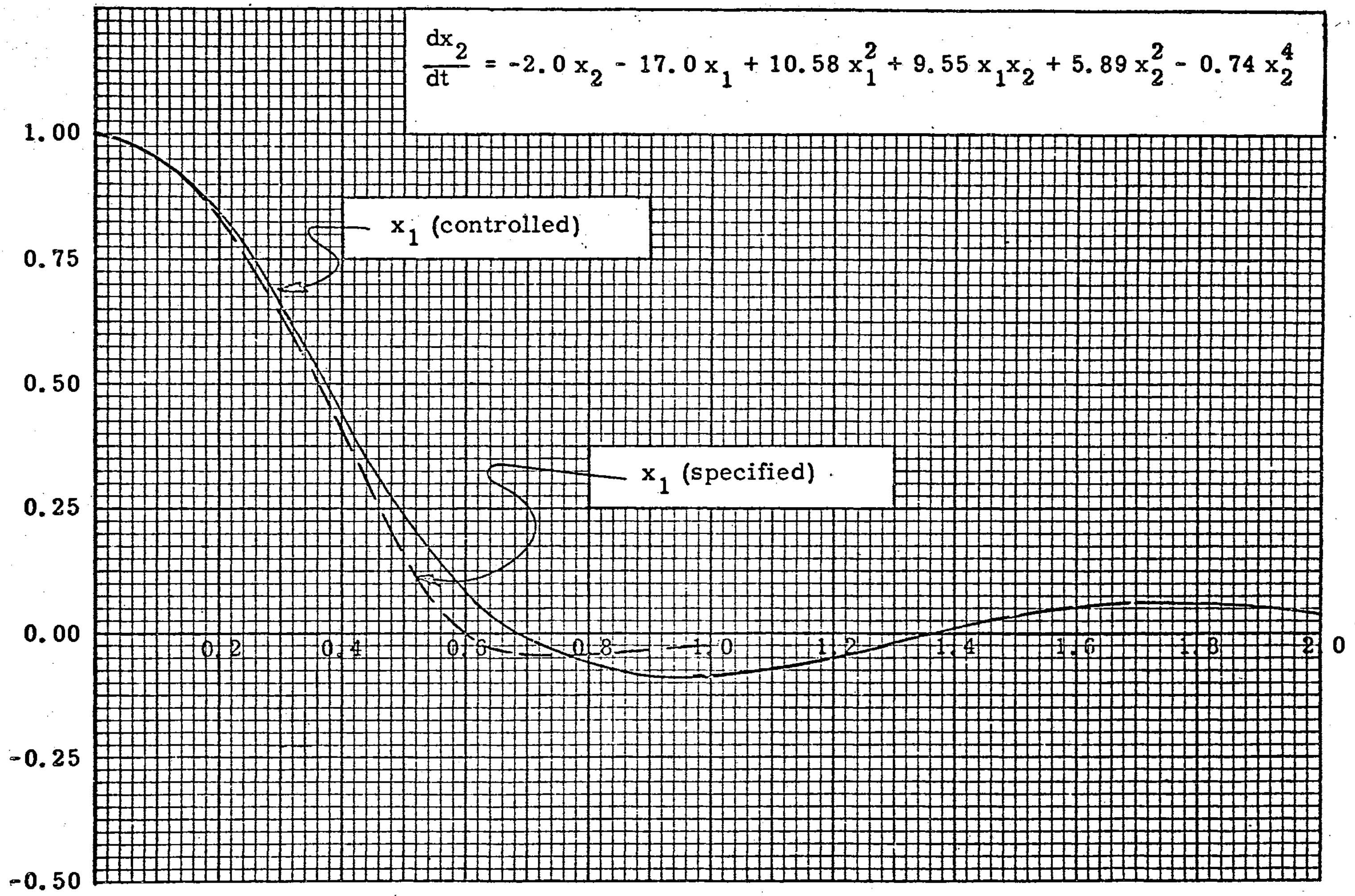


Figure 2-1. Controlled Response for Overdamped Linear Oscillator

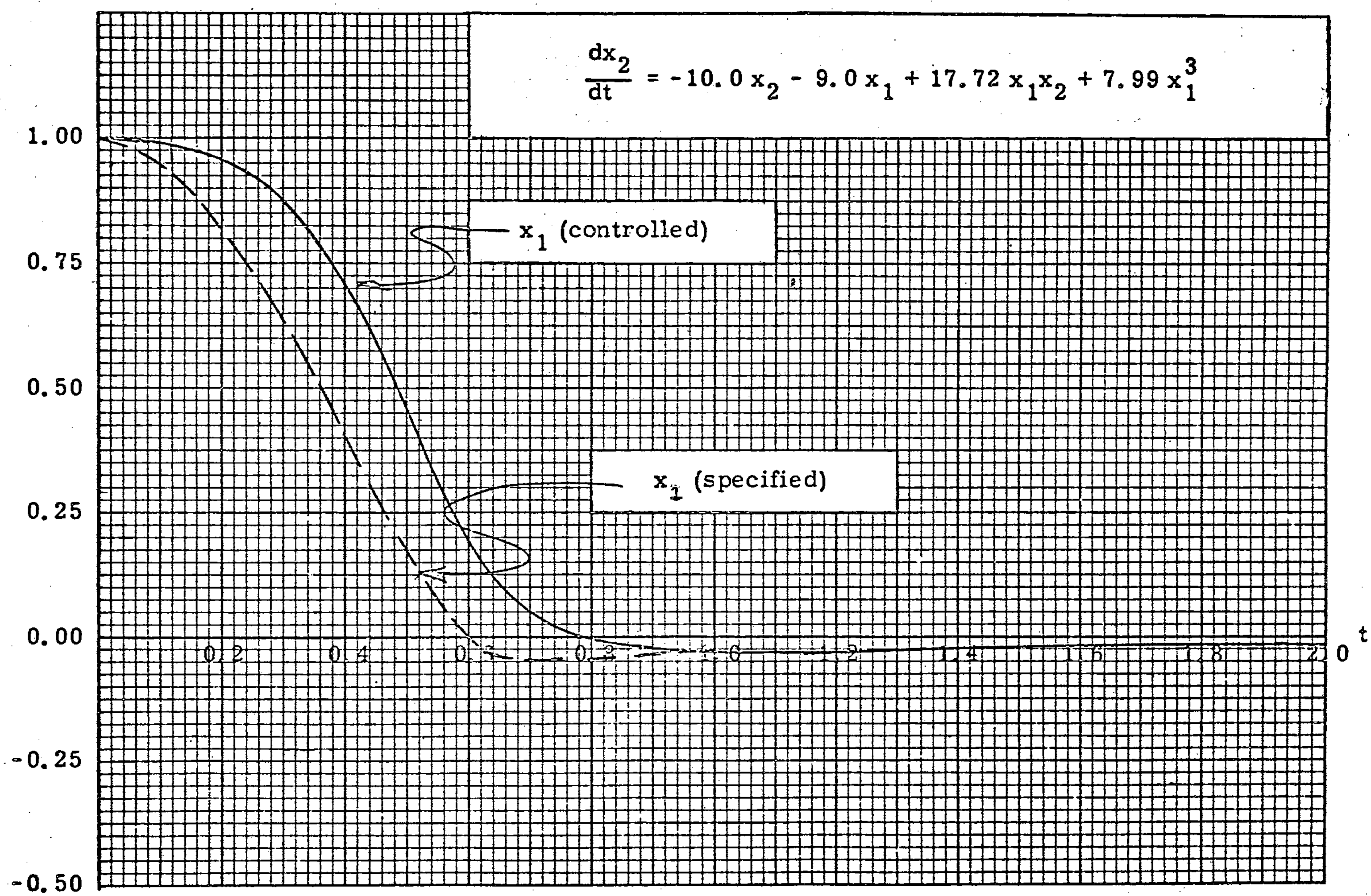


Figure 2-2. Controlled Response for Underdamped Linear Oscillator

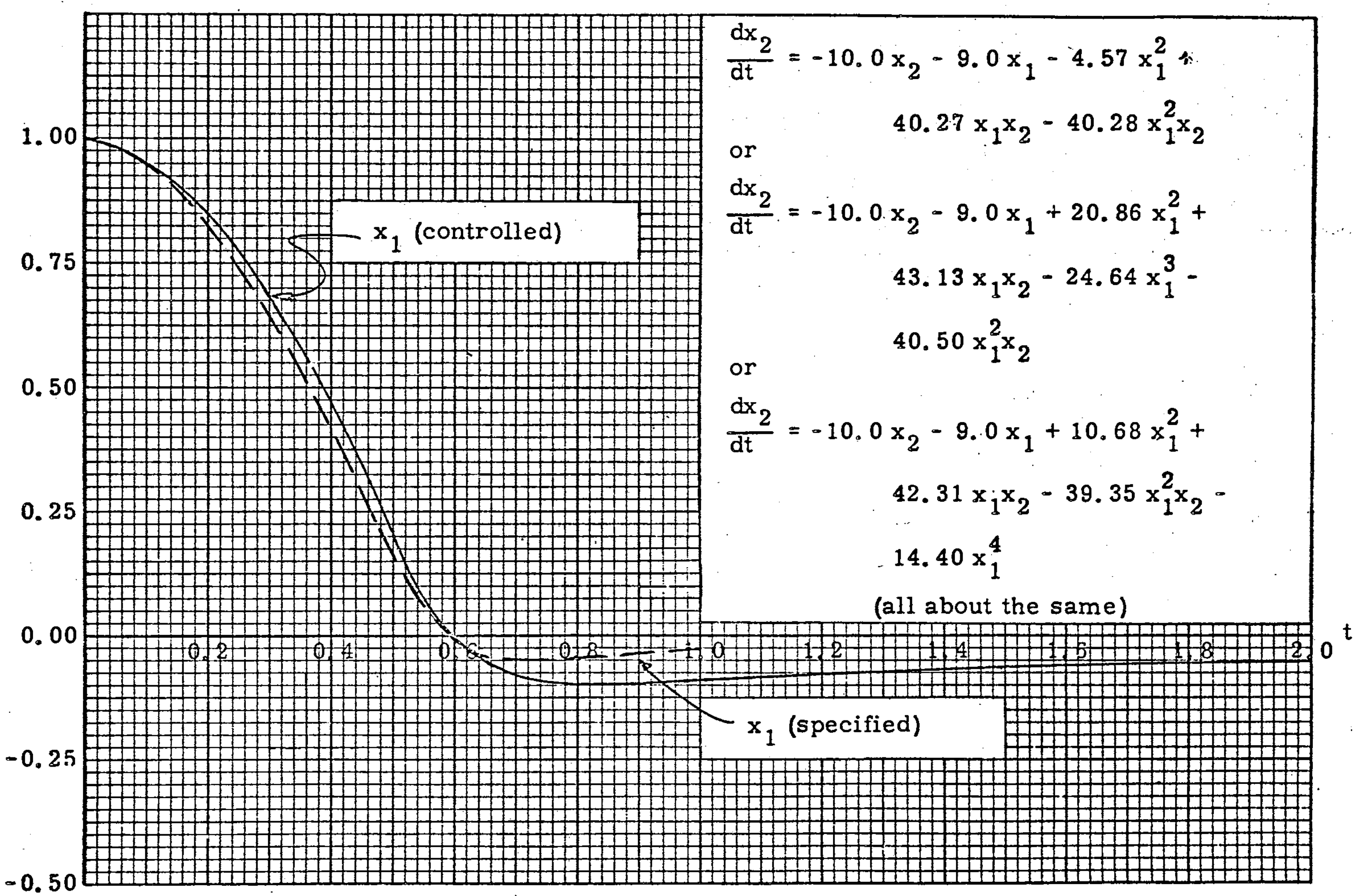


Figure 2-3. Controlled Response for Overdamped Linear Oscillator

(2) The state variables X_2 and $\dot{X}_2$ were obtained by differentiating an arbitrarily drawn response. This led to an introduction of errors in the specification of the system.

The dynamic equation which produced a response almost exactly like the specified transient responses of Figures 2-1 through 2-3 contained eight nonlinear terms. The control equation, while forcing the system to conform to the specified response, was quite complicated. This complicated control equation was a result of specifying the response for all values of time. In order to decrease the complexity of the system equation, the response characteristics in this thesis will be forced to satisfy a set of essential system characteristics and not a continuous time response. By having this latitude, a number of different sets of state variable representations can be used in order to fit a desired system equation to a specified transient response. The fitting of a response by the least squares technique requires computation of the velocity and acceleration with respect to time corresponding to the displacement response. In addition, the assumed form of the differential equation becomes complicated in order to satisfy the constraints imposed by a continuous time response.

The preceding discussion summarizes past reports on methods of forcing or controlling a system to meet a desired response. In particular, the works of Bernhart and Hove indicate the feasibility of using nonlinear components in a system to improve performance. Hove showed that a desired performance specification can be met if a sufficient number of nonlinear terms are used. However, many of the terms required cannot be realized as physical hardware. In some cases, the identification of the control vector g_i for a physical design is questionable. The difficulty in using the method proposed by Hove is in obtaining a physical system which will conform to a continuous response. This difficulty will be partially reduced by specifying only a set of essential system characteristics and not a continuous response.

CHAPTER III

THE ANALYSIS METHOD

The aim of the present work is to develop a procedure for improving the response characteristics of a dynamic system whose performance requirements are specified. The approach used is:

(1) to specify the required system response to a step disturbance in terms of rise time, overshoot, and settling time and

(2) to add a minimum number of physical attainable nonlinear functions to the system equation to achieve the specified response performance. The basic necessity is to establish the control vector g_i.

The method used to fit the dynamic equation to the desired specification involves a least squares fitting of a set of state variables of a system to a predetermined control vector made up of a set of physically relizable control elements. The functional form of the control vector will be specified and the dynamic system will be forced to satisfy a set of essential system characteristics. The state variables of the system are defined in terms of a straight line approximation. The desired specification is defined in terms of the response of the system to a unit step input function. A hydraulic control system is used as an illustrative example.

Development of the Least Squares Method

The mathematical models describing the behavior of a physical system consist of a set of ordinary differential equations with fixed coefficients of the form

$$a_n(X)\,\frac{d^nX}{dt^n} + a_{n-1}(X)\,\frac{d^{n-1}X}{dt^{n-1}} + \dots + a_0(X) = F(t). \tag{3-1}$$

It is often desirable to force such a system to perform in a manner dictated by predesignated performance criteria. The criteria usually infer an "optimum" performance based on a set of physical system constraints. Compensation of the system described by Equation (3-1) requires determination of the changes or additions to the physical system which must be made to obtain the desired response.

To implement the analysis method, the physical system of n^{th} order is described at any time t by means of the finite set of quantities $X_1(t), X_2(t), \dots, X_n(t)$. For example, these variables might represent the position, velocity, acceleration, etc., of a physical system. These quantities are referred to as the state variables of the system and define the components of the state vector $\mathbf{X}(t)$ of the system. The time changes in the system are related to the state of the system by assuming that the derivative of the state vector, dx/dt, depends only upon the current state of the system and not upon its past history. Using this basic assumption leads to the mathetmatical representation of a system by means of a vector-matrix differential equation of the form

$$\dot{X}(t) = \frac{dx(t)}{dt} = f[X(t),\, g(t),\, t] \tag{3-2}$$

with initial conditions $x(0) = x_0$. In Equation (3-2), $g(t)$ represents a control vector made up of a finite number of control parameters, and f is a vector function of the state variables, the control signals, time, and possibly the environmental variables or external disturbances. The control vector g must satisfy basic system requirements which reflect the restrictions imposed upon the control system.

The problem of design of a control system may be stated as follows: Given a system of the form

$$\frac{dX_i}{dt} = f_i(X_i) \tag{3-3}$$

to be controlled, determine the coefficients a control vector g_i, of predetermined functional form, such that the system

$$\frac{dX_i}{dt} = F_i(X_i,\ g_i) \tag{3-4}$$

is controlled in such a way that a set of performance indices is optimized.

The performance indices might include system constraints on any or all of the system's state variables. For example, if the system's state variables are displacement, velocity, and acceleration, then constraints on system velocities in a hydraulic system could be imposed in order to avoid harmful cavitation (6).

In the analysis procedure, it is assumed that the system's state can be controlled; that is, there is system control vector g_i whose manipulation governs the system state. The control vector g_i is given by

$$g_i = g_i(u_i) \tag{3-5}$$

is made up of a set of physically realizable control parameters u_{il} of the form

$$u_{il} = u_{il}(X_1, X_2, \ldots, X_N) \tag{3-6}$$

By adjoining this set of physical realizable control parameters to the fixed or uncontrolled system, the behavior of the dynamic system performance is characterized by a set of system differential equations of the form

$$\frac{dX_i}{dt} = f_i(X_1, X_2, \ldots, X_N) + g_i(X_1, X_2, \ldots, X_N) \tag{3-7}$$

where f_i represents the original system and g_i is the vector control function used to achieve a predetermined system performance.

The dynamic system represented by Equation (3-7) may be expressed in the following form:

$$\begin{aligned} \frac{dX_i}{dt} &= K_{ij}U_{ij} + a_{il}u_{il} \\ j &= 1, 2, \ldots, P_i \\ l &= 1, 2, \ldots, q_i \\ i &= 1, 2, \ldots, N \end{aligned} \tag{3-8}$$

The original system terms $K_{ij}U_{ij}$, $j = 1, 2, \ldots, P_i$ are of fixed form and their magnitudes are determined by mathematically modeling the fixed physical system.

The functional form of the control vector

$$g_i = g_i(a_{il}u_{il}) \quad , \quad l = 1, 2, \ldots, q_i \tag{3-9}$$

is assumed to be known and the coefficients are to be determined in the analysis procedure. The terms which make up the control vector are made up of a set of physically realizable control parameters which are defined for a specific system. The analysis method, assuming that the form of the control vector has been established is to determine the magnitude of the coefficients a_{il}. The determination of the magnitude of these coefficients is based on a least squares fitting technique; that is, the analysis procedure is to determine the a_{il} such that the sum of the squares of the difference between the designed system variables and the specified system variables is a minimum. In the analysis method, the differential equation expressed by Equation (3-8) is

$$Y_i - a_{il}u_{il} = 0 \tag{3-10}$$

where

$$Y_i = \frac{dX_i}{dt} - K_{ij}U_{ij}.$$

The left side of Equation (3-10) is the fixed or original system while the right side is variable since the a_{il} can be changed. The analysis method consists of selecting m points in time from the state variable representation of the system and solving this set of m equations by a least squares fitting approach. The least squares procedure requires that the a_{il} be chosen in such a way that the difference between the fixed and variable parts of Equation (3-10), the error or residue, at each point m is a minimum; that is,

$$\sum_{k=1}^{m} (Y_i - a_{i1}u_{i1})_k^2 = \sum_{k=1}^{m} e_k^2. \qquad (3\text{-}11)$$

In statistics, the residual e given in Equation (3-11) is commonly referred to as a random variable. The least squares method is used to find the value of a_{i1} such that the sum of squares $\sum_{k=1}^{m} e_k^2$ is a minimum. With a, u, Y, and e in vector form, this gives

$$\sum_{k=1}^{m} e_k^2 = e'e = (Y - au)'(Y - au) \qquad (3\text{-}12)$$

where e' is the transpose of e. The value of a that minimizes $e'e$ is given by the solution to

$$\frac{\partial}{\partial a} (e'e) = 0. \qquad (3\text{-}13)$$

The solution to Equation (3-13) results in the matrix equations

$$u'ua = u'Y \qquad (3\text{-}14)$$

called the normal equations. The solution of the normal equations results in the least squares estimate of a_{i1} defined as

$$\hat{a} = S^{-1}u'Y \qquad (3\text{-}15)$$

where

$$S^{-1} = (u'u)^{-1}.$$

The elements of the column matrix a_{i1} are the values of the unknown coefficients of the control vector g_i defined by Equation (3-9).

Equation (3-10), sometimes referred to as the prediction equation, may be expressed in vector form as

$$\underset{m\times 1}{Y} = \underset{m\times q}{u}\ \underset{q\times 1}{a} + \underset{m\times 1}{e} \tag{3-16}$$

where

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \cdot \\ \cdot \\ \cdot \\ Y_m \end{bmatrix} \tag{3-17}$$

$$u = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1q} \\ u_{21} & u_{22} & \cdots & u_{2q} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ u_{m1} & u_{m2} & \cdots & u_{mq} \end{bmatrix} \tag{3-18}$$

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_q \end{bmatrix} \tag{3-19}$$

$$e = \begin{bmatrix} e_1 \\ e_2 \\ \cdot \\ \cdot \\ \cdot \\ e_m \end{bmatrix} \tag{3-20}$$

Examining $Y = au + e$, it can be seen that for each set of u's, u_{11}, u_{12}, ..., u_{iq} that are selected, a corresponding value of Y must be chosen. Additional sets of u's and Y's are then chosen until m values of the u's and Y's have been chosen. The elements of the normal equations defined by Equation (3-14) are given below

$$\begin{bmatrix} \Sigma u_{i1}^2 & \Sigma u_{i1}u_{i2} & \cdots & \Sigma u_{i1}u_{iq} \\ \Sigma u_{i2}u_{i1} & \Sigma u_{i2}^2 & \cdots & \Sigma u_{i2}u_{iq} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \Sigma u_{iP}u_{i1} & & \cdots & \Sigma u_{iq}^2 \end{bmatrix} \begin{bmatrix} a_{i1} \\ a_{i2} \\ \cdot \\ \cdot \\ a_{iq} \end{bmatrix} = \begin{bmatrix} \Sigma u_{i1}Y_i \\ \Sigma u_{i2}Y_i \\ \cdot \\ \cdot \\ \Sigma u_{iq}Y_i \end{bmatrix} \qquad (3\text{-}21)$$

where the summation is taken over the m sets of data.

Verification of the Least Squares Procedure

The fitting of a differential equation to a system requires that the form of the differential equation be specified and also that the time history of the state variables of the system be specified. The form of the differential equation depends upon the physical system, and the time history of the state variables of the system depends upon the desired system characteristics. Before any discussion about the system characteristics or differential equation form is made, a check of the validity of the least squares procedure will be made.

To check the validity of the least squares procedure, the following differential equation was assumed

$$\ddot{X} + B\dot{X} + CX + DX^3 + EX\dot{X} = 1. \qquad (3\text{-}21a)$$

In the first check, the state variables for the system were obtained by assigning values for B, C, D, and E and then solving this nonlinear differential equation by the Runge-Kutta method (see Appendix A) to obtain the displacement, velocity, and acceleration of the system. These state variables were then used to generate the normal equations defined by the matrix (3-21). The matrix representation of the normal equations for Equation (3-21a) is

$$\begin{bmatrix} \Sigma \dot{X}_i^2 & \Sigma X_i \dot{X}_i & \Sigma \dot{X}_i X_i^3 & \Sigma X_i \dot{X}_i^2 \\ \Sigma X_i \dot{X}_i & \Sigma X_i^2 & \Sigma X_i^4 & \Sigma X_i^2 \dot{X}_i \\ \Sigma \dot{X}_i X_i^3 & \Sigma X_i^4 & \Sigma X_i^6 & \Sigma X_i^4 \dot{X}_i \\ \Sigma X_i \dot{X}_i^2 & \Sigma X_i^2 \dot{X}_i & \Sigma X_i^4 \dot{X}_i & \Sigma X_i^2 \dot{X}_i^2 \end{bmatrix} \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix} = \begin{bmatrix} \Sigma \dot{X}_i (\ddot{X}_i - 1) \\ \Sigma X_i (\ddot{X}_i - 1) \\ \Sigma X_i^3 (\ddot{X}_i - 1) \\ \Sigma X_i \dot{X}_i (\ddot{X}_i - 1) \end{bmatrix} \tag{3-21b}$$

where the summation is taken over m sets of data points. The coefficients B, C, D, and E which were used to generate the state variables of the system and the coefficients obtained by the least squares fitting of this data are given in Table I. The data was sampled at every tenth of a second for 15 seconds, making a total of 150 data points.

TABLE I

DIFFERENTIAL EQUATION COEFFICIENTS

Coefficient	Assumed Coefficients	Computed Coefficients
B	5.60272	5.603
C	0.30911	0.3091
C	0.69954	0.6996
D	0.69954	0.6996
E	-4.64541	-4.645

In the second check, the two coefficients B and C were set equal to 5.60272 and 0.30911, respectively. The matrix representation of the normal equations for this system with the two fixed coefficients B and C is

$$\begin{bmatrix} \Sigma X_i^6 & \Sigma X_i^4 X_i \\ \Sigma X_i^4 \dot{X}_i & \Sigma X_i^2 \dot{X}_i^2 \end{bmatrix} \begin{bmatrix} D \\ E \end{bmatrix} = \begin{bmatrix} \Sigma X_i^3 (\ddot{X}_i + B\dot{X}_i + CX_i - 1) \\ \Sigma X_i \dot{X}_i (\ddot{X}_i + B\dot{X}_i + CX_i - 1) \end{bmatrix} \qquad (3\text{-}21c)$$

The values of coefficients D and E were 0.69954 and -4.64556, respectively.

In the third check, the state variables of a second-order linear differential equation were used as input data for the normal equation given by Equation (3-21) to determine if the least squares procedure would recognize the data as being linear and set the coefficients of the nonlinear terms equal to zero. The linear differential equation used to generate the state variables of the system was

$$\ddot{X} + 0.8\dot{X} + X = 1 \qquad (3\text{-}21d)$$

The nonlinear differential equation which was to be fitted to this data is given by Equation (3-21a). The least squares procedure returned the following nonlinear differential equation

$$\ddot{X} + 0.8000\dot{X} + 1.000X + 0.0000002608X^3 - 0.0000007153X\dot{X} = 1 \qquad (3\text{-}21e)$$

The coefficients of the linear portion of the equation were within four significant digits of Equation (3-21d). The magnitude of the coefficients of the nonlinear terms were small. The small error can be attributed to round-off error in the numerical procedure.

These three exercises confirm that the least squares fitting of a differential equation can be made, given the state variables of the system. In the following section, a method of obtaining the state variables of a system by defining a set of performance characteristics is presented.

Performance Specifications

In this section, a method of specifying a set of performance characteristics for a second-order system is presented. The method makes use of a set of straight lines which approximate the displacement, velocity, and acceleration of the system in terms of the transient response to a unit step function input. The desired system characteristics are defined in terms of the transient response characteristics: (1) overshoot and time to overshoot, (2) rise time, and (3) settling time. From these system characteristics, a complete time history of the state variables which describe the system are derived.

The straight-line approximations of the displacement, velocity, and acceleration are the input data for the least squares fitting method. The least squares method fits the discontinuous straight-line approximations to a continuous differential equation. The type of differential equation to which the straight-line approximations are fitted depends on the nature of the desired unit step response. The type of responses to be considered in this thesis will be modeled by nonlinear second-order differential equations with constant coefficients; that is, equations of the form

$$\ddot{X} + h(X, \dot{X}) = 1 \qquad (3\text{-}22)$$

The least-squares method determines the magnitude of the coefficients of the differential equation. This equation is then solved by a Runge-Kutta method (see Appendix A) to obtain the fitted continuous response.

The least squares synthesis procedure requires that the state variables be specified. For a second-order system, this requires a time history of the displacement, velocity, and acceleration. In most cases, a continuous plot is not specified but only a set of essential characteristics of the transient response of the system. By defining the desired performance characteristics in this manner, some latitude in the design is available and some of the important features of the system may be emphasized.

The essential characteristics of a second-order system are specified in terms of the transient response to a unit step function input. A basis for evaluating the performance of a system is in terms of the following quantities and is represented graphically in Figure 3-1.

The defined characteristics of the response shown in Figure 3-1 are:

(1) MAXIMUM OVERSHOOT, OS, is the maximum value of the time response.

(2) TIME TO MAXIMUM OVERSHOOT, T_{OS}, is the time required to reach the maximum overshoot.

(3) RISE TIME, T_{RT}, is the time required to reach the final value the first time.

(4) SETTLING TIME, T_{ST}, is the time required for the oscillations to die down to the specified absolute

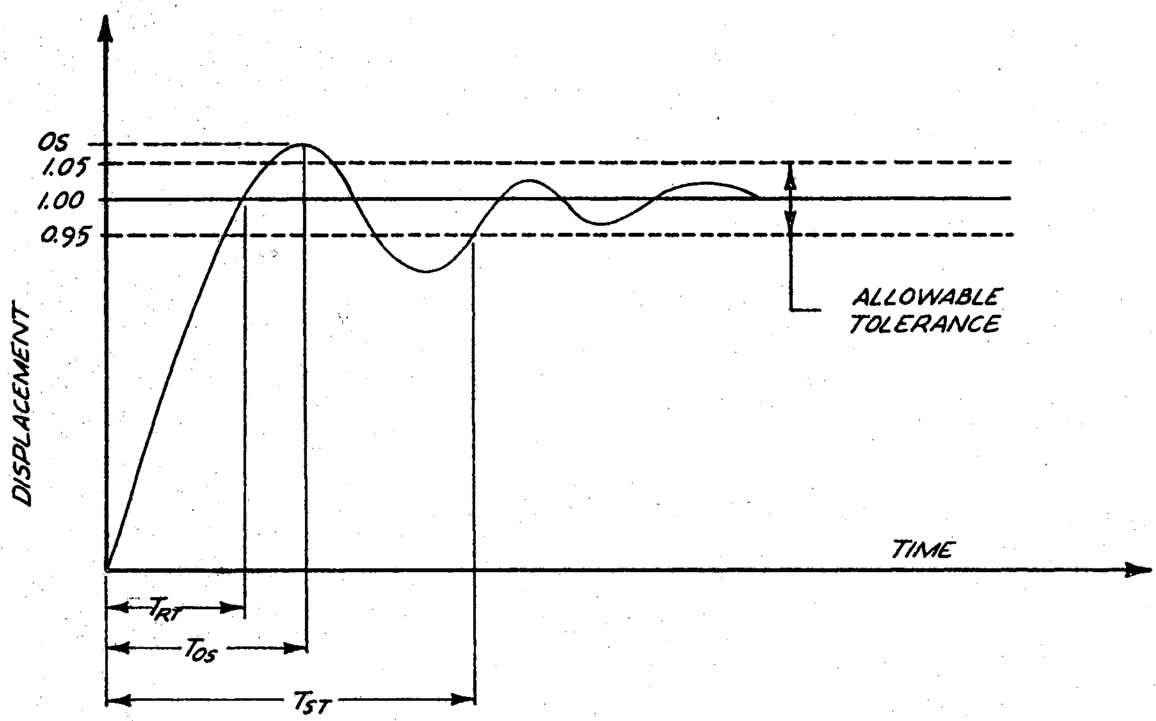


Figure 3-1. Second-Order Transient Response

percentage of the final value and remain less than this value. This percentage must be specified in the individual case. Common values used for settling time are two and five per cent.

In order to make a reasonable comparison between various systems, it is necessary to start with standard initial conditions. The most practical standard is to start with the system at rest. Then the response characteristics, such as maximum overshoot, settling time, etc., can be compared.

The information given by specifying the essential characteristics of a system as shown in Figure 3-1 is incomplete in defining a specific transient response. A large number of system responses will satisfy the set of essential characteristics; however, a given physical system may be restrained such that only a few specific responses are possible. These restraints may be in the form of maximum velocities or accelerations.

The approximation method in the next section provides a means by which the essential characteristics of a system response may be met when constraints are imposed on system variables.

The Approximation Method

The set of essential characteristics which define a desired system response are defined in Figure 3-2. The specified terms are rise time, overshoot, time to overshoot, and settling time. These system characteristics must be met in specifying the time history of the system. The desired system response must rise and cross the final value in a time equal to or less than that defined by point A_1 without overshooting

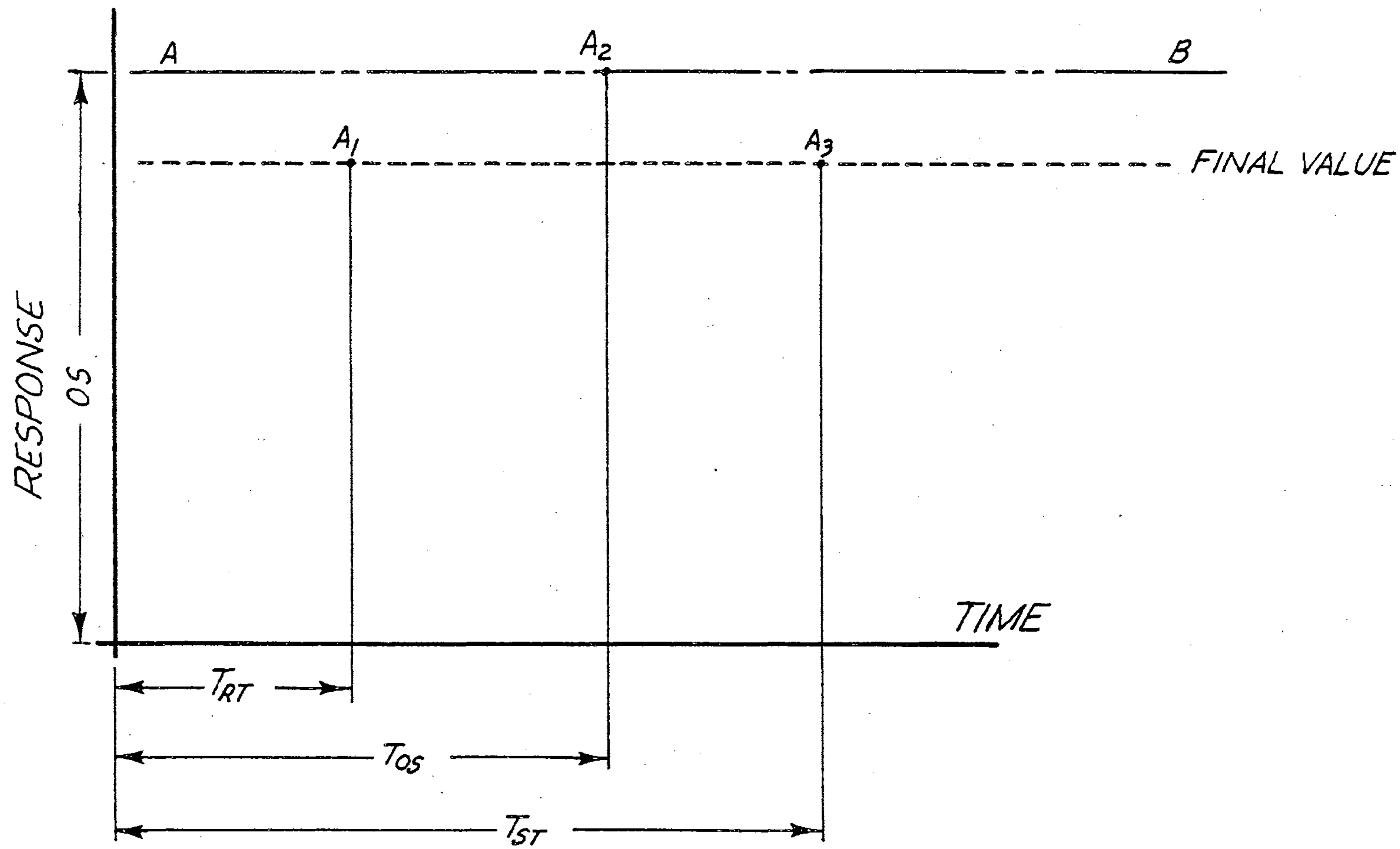


Figure 3-2. Essential System Characteristics

above the boundary line AB. In addition to this, the system must settle to within a certain percentage of the final value within the time T_{ST}.

Any number of curves may be fitted through these points and boundaries and still satisfy the requirements defined in Figure 3-2. With this latitude, it is possible to fit a system response to a set of desired system characteristics and still meet other constraints which may be imposed on the system. Figure 3-3 illustrates this point by showing two different types of responses with the same essential system characteristics but with different maximum velocities and accelerations.

From the characteristics defined by Figure 3-2, a set of state variables (displacement, velocity, and acceleration) such as those illustrated in Figure 3-3 must be obtained. The proposed way to obtain the state variable of the system is to approximate the displacement, velocity, and acceleration by a set of straight lines and then to smooth these straight lines by the least squares method defined in the previous section. The straight-line approximation for a set of compatible variables for a second-order system is illustrated by Figures 3-4, 3-5, and 3-6. These approximations of the state variables of the system are functions of the characteristics defined in Figure 3-2. In the first evaluation the values T_1, T_2, and T_3 are equal to the rise time, time to overshoot, and settling time, respectively. The value OS is equal to the step value plus the overshoot value.

The velocity and acceleration of the system deviate from the true differentiation of the displacement approximation. These deviations produce area errors which must be corrected. These corrections are discussed in the next section.

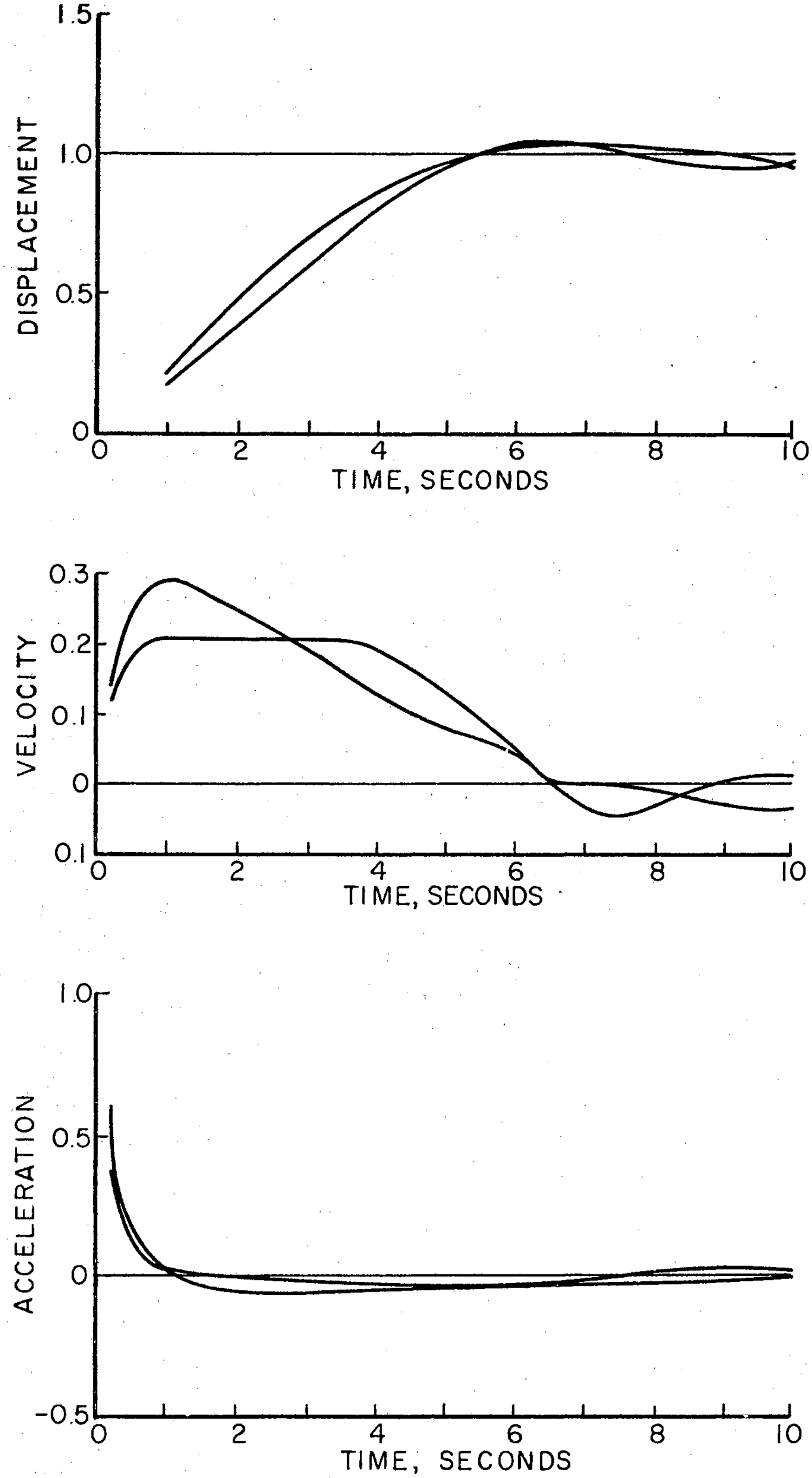
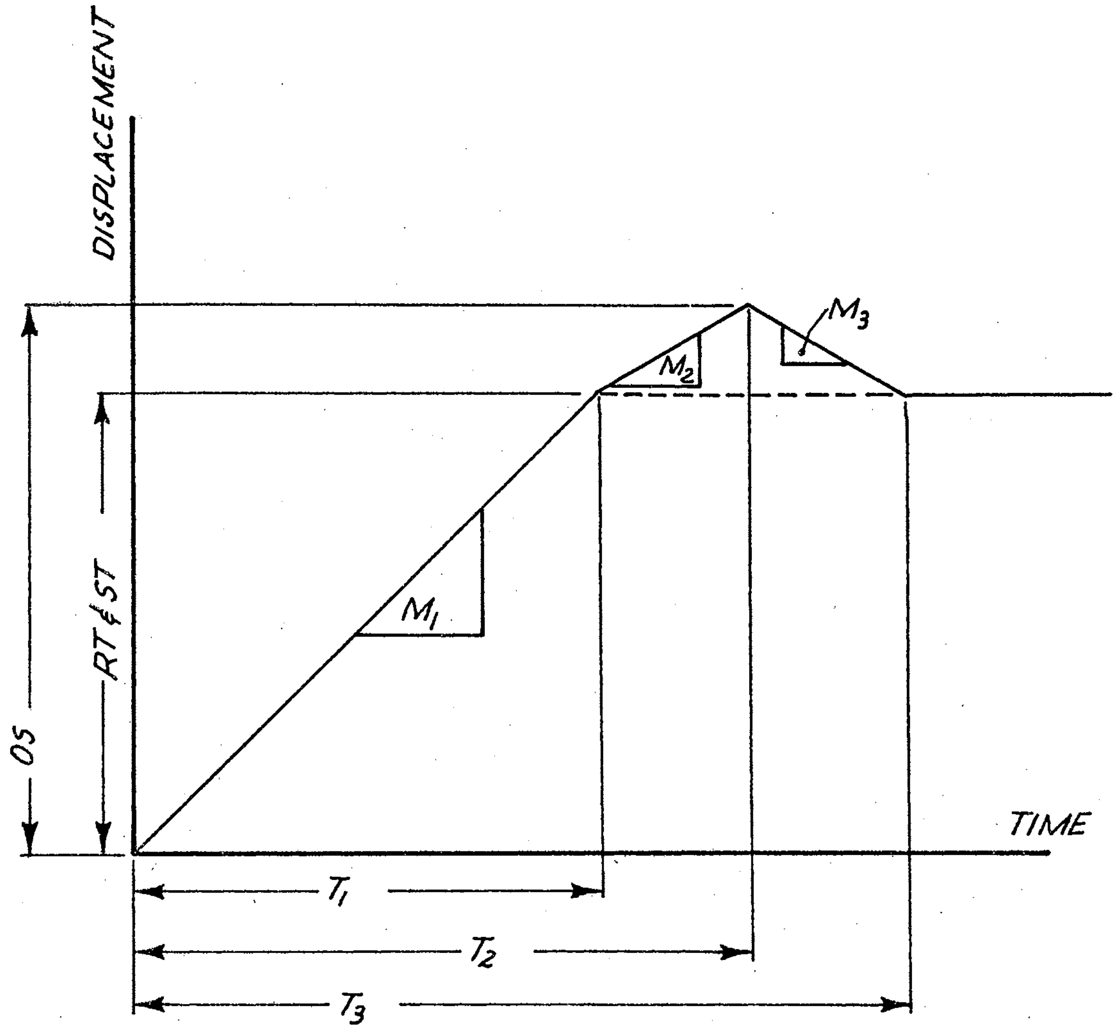
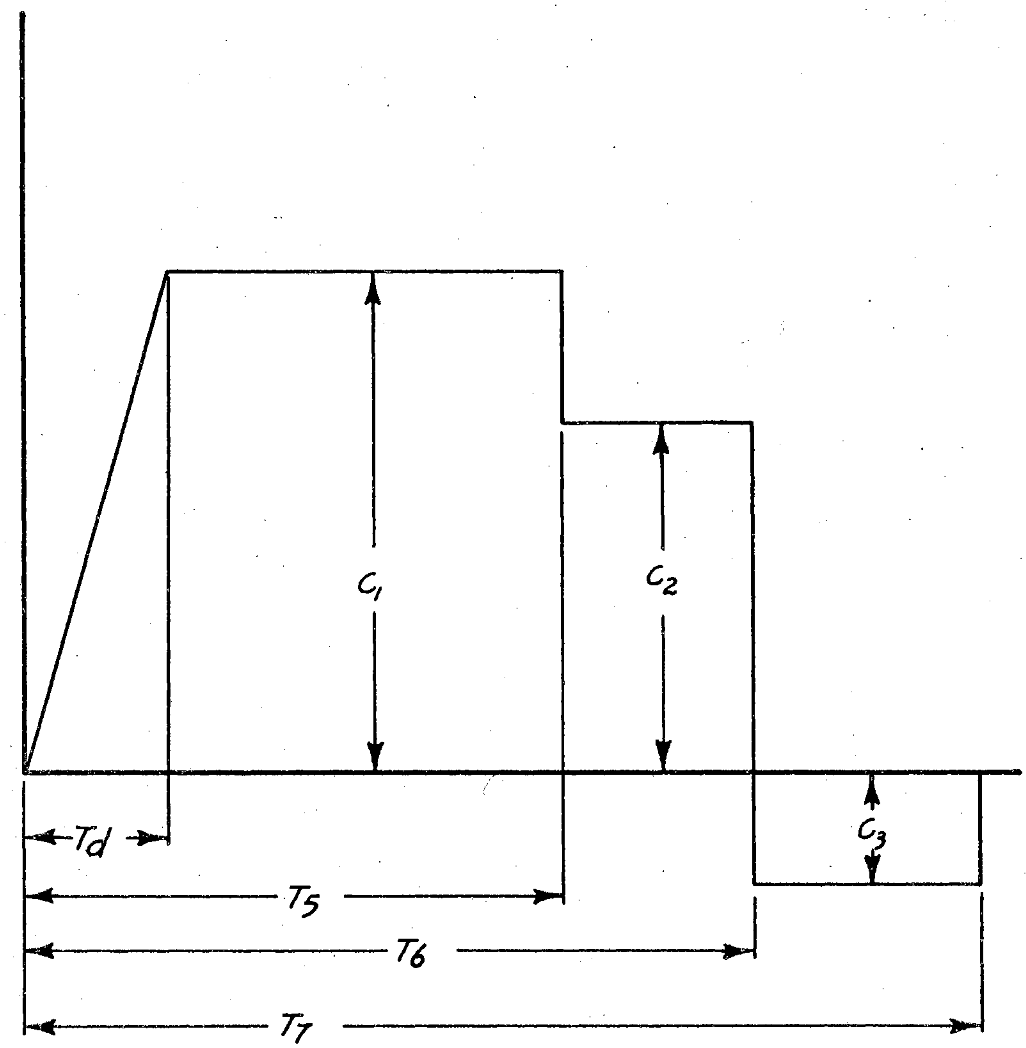


Figure 3-3. Second-Order System State Variables



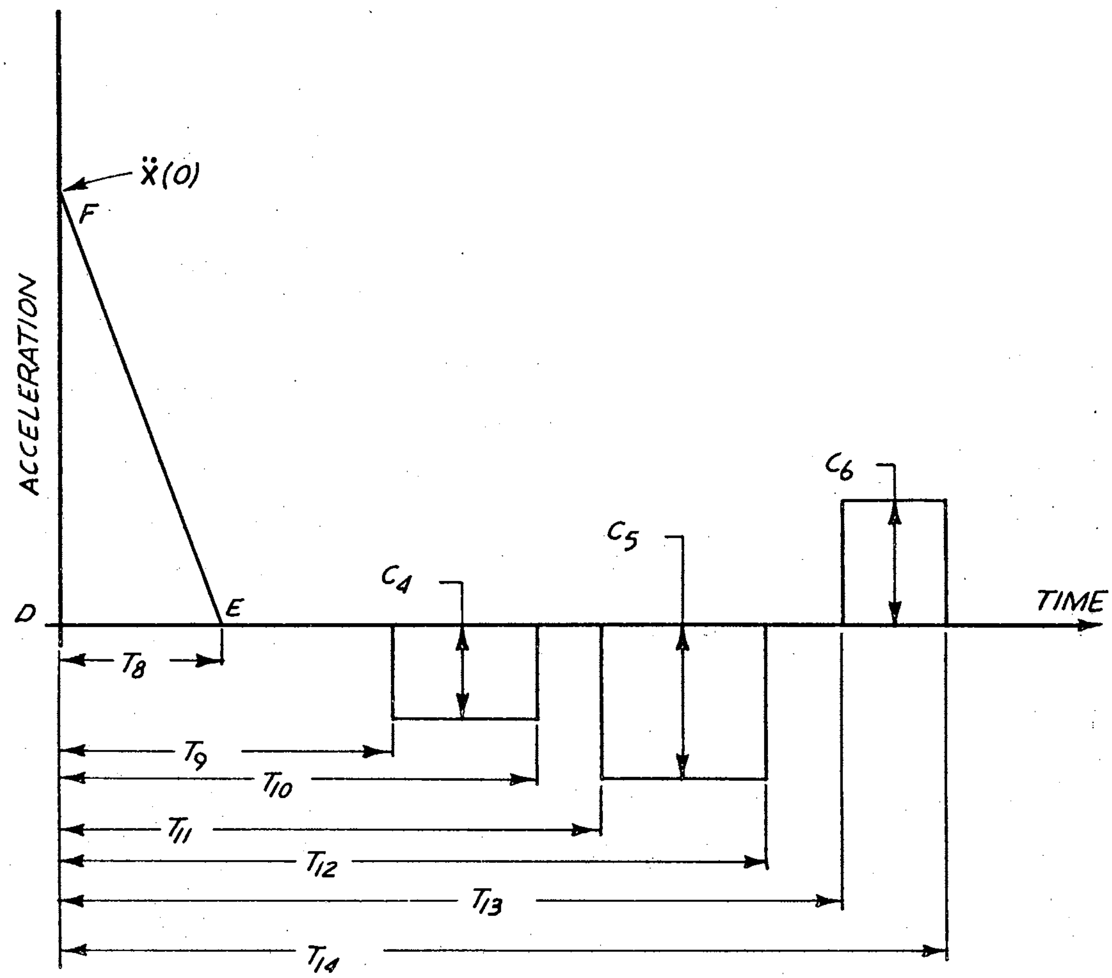
T_1 = TIME TO FIRST CROSSING
T_2 = TIME TO OVERSHOOT
T_3 = TIME TO SETTLING TIME
OS = DISPLACEMENT MAXIMUM
RT & ST = FINAL VALUE
$M_1 = RT/T_1$
$M_2 = (OS-RT)/(T_2-T_1)$
$M_3 = (ST-OS)/(T_3-T_1)$

Figure 3-4. Displacement Approximation



$T_5 = T_{RT}$
$T_6 = T_{OS}$
$T_7 = T_{ST}$
$C_1 = 1/[T_1 - \frac{1}{2}(T_4)]$
$C_2 = M_2$
$C_3 = M_3$

Figure 3-5. Velocity Approximation



$T_8 = T_d(T_1)$
$T_9 = T_1 - C_8(T_2 - T_1)$
$T_{10} = T_1 + C_8(T_2 - T_1)$
$T_{11} = T_2 - C_8(T_3 - T_2)$
$T_{12} = T_2 + C_8(T_3 - T_2)$
$T_{13} = T_3 - C_8(T_3 - T_2)$
$T_{14} = T_3$

$C_4 = (C_0 - M_2)/(T_9 - T_{10})$

$C_5 = (M_2 - M_3)/(T_{11} - T_{12})$

$C_6 = -M_3/(T_{14} - T_{13})$

$\ddot{X}(0) = C_0/[0.5(T_4)]$

Figure 3-6. Acceleration Approximation

Discussion of Straight-Line Method

The characteristics defined in Figure 3-2 are used to implement the displacement approximation. Since no prior knowledge about the form of the displacement response is available at this point, straight lines are used to construct the displacement curve. Straight lines are drawn from the origin to the final value at a time equal to T_{RT}, from the final value to the maximum overshoot at a time equal to T_{OS}, and from the maximum to the final value at a time equal to T_{ST}, as shown in Figure 3-4. If T_{OS} is not specified, a value equal to $T_{RT} + \frac{1}{2}(T_{ST} - T_{RT})$ is assumed. The exact form of the displacement response from $t = 0$ up to the first crossing is a function of the form of the velocity curve.

Differentiating the displacement curve defined in Figure 3-4 results in a velocity curve (see Figure 3-7) which meets the area requirement but not the initial value requirement. T_d, defined in Figure 3-7, is introduced to meet the initial value requirement and to provide a means of shaping a desired velocity curve by delaying the time that the velocity reaches its maximum value. The changes in the system velocity as a result of the addition of the delay time T_d are shown graphically in Figure 3-7.

Defining the delay time T_d results in an area relationship error. This error must be compensated for by increasing the value M_1 such that the area enclosed by the velocity curve up to the time $t = T_{RT}$ must be equal to the final value.

This corrected value of the velocity, C_0, is determined by equating the area of the velocity curve up to the time $t = T_{RT}$ equal to the final value. Mathematically, this is

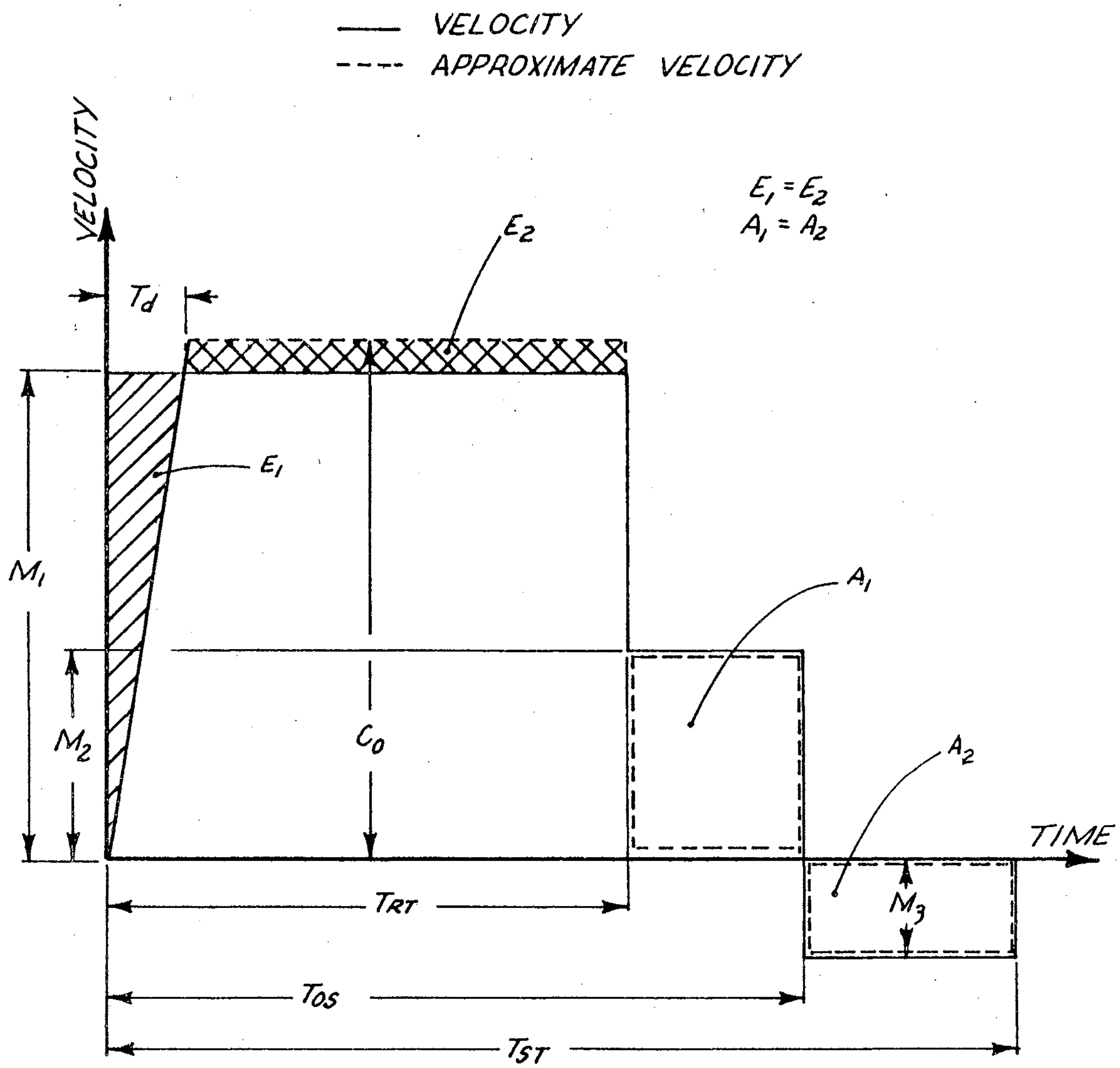


Figure 3-7. Initial and Modified Velocity Approximations

$$\tfrac{1}{2}(T_d)C_O + (T_{RT} - T_d)C_O = 1.0 \tag{3-23}$$

or

$$C_O = 1/[T_{RT} - 0.5(T_d)] \tag{3-24}$$

The value C_O is the corrected value of the velocity from time $t = T_d$ to time $t = T_{RT}$. The area enclosed by the curve from time T_{RT} to T_{ST} must be equal to the overshoot value. This area must also be equal to that enclosed by the velocity curve from T_{OS} to T_{ST}. The summation of the total area at time T_{ST} and greater must be equal to the final value of the displacement. The values M_2 and M_3 are the exact derivatives obtained from the displacement approximation.

The integrated sum of the acceleration curve along the time axis must be equal to the velocity curve. The total area of the acceleration curve up to the time $t = T_d$ must be equal to the value of the corrected velocity C_O. Since T_d (see Figure 3-7) is fixed, $\ddot{X}(0)$ is given a value such that the area of the triangle DEF equals C_O. If $\ddot{X}(0)$ is made equal to

$$X(0) = C_O/[0.5(T_d)] \tag{3-25}$$

the area relationship for the acceleration curve from time $t = 0$ to $t = T_d$ is satisfied. The approximate velocity curve makes an abrupt change at time $t = T_{RT}$, $t = T_{OS}$, and $t = T_{ST}$. These abrupt changes are "averaged out" by extending the time interval over which the area relationship must be satisfied. For example, the velocity curve must change from a value of M_1 to C_O in zero time in order to satisfy the velocity curve. This area requirement, given by the difference between M_2 and C_O,

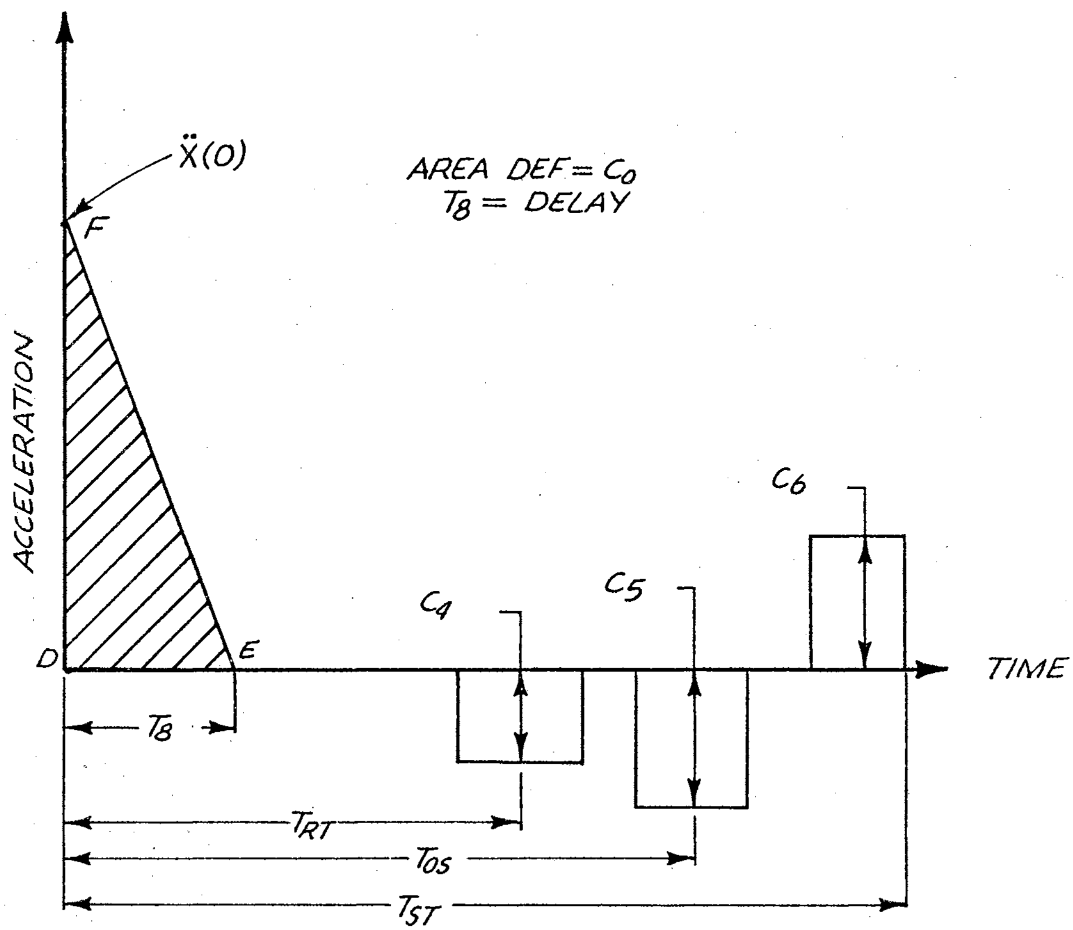


Figure 3-8. Modified Acceleration Approximation

is averaged out over the time interval from $T_9 = T_{RT} - C_8(T_{OS} - T_{RT})$ to $T_{10} = T_{RT} + C_8(T_{OS} - T_{RT})$. The extension of the time interval from $T = 0$ to $T_{10} - T_9$ will require that the value of C_4 be equal to

$$C_4 = (C_0 - M_2)/(T_{10} - T_9). \tag{3-26}$$

The value of C_8 which shrinks or extends the time interval must fall within the limits $0.1 \leq C_8 \leq 0.5$.

The time intervals and values for C_5 and C_6 are obtained in the same manner and are given in the definition figure of the acceleration (see Figure 3-6). The total area at time $t = T_{ST}$ and greater must be equal to the final value of the velocity.

The straight lines of the approximated state variables of the system are used as input data to the least squares fitting method. The least squares method is used to "average out" the discontinuities of the straight-line approximation. The block diagram for generating the straight lines and fitting a differential equation is given in Figure 3-9.

Example

In order to demonstrate the analyses method, the following assumed differential equation

$$\ddot{X}_1 + A_1\ddot{X} + B_1X + C_1X^3 + D_1X\dot{X} = 1 \tag{3-27}$$

is fitted to the following set of performance specifications:

(1) Rise time = 5.0 seconds

(2) Per cent overshoot = 6

(3) Settling time = 7.5 seconds.

Specify

(1) Rise Time
(2) Overshoot
(3) Time to Overshoot*
(4) Time to Settling Time
(5) Assume $T_d = 0.1$*

↓

Generate the displacement, velocity, and acceleration curve from the definition curves of Figure 3-4, 3-7, and 3-8.

↓

Select a smoothing differential equation of the form

$\ddot{X} + h(X, \dot{X}) = 1.$

↓

Generate the "Normal Equations" defined by Equation 3-21.

↓

Solve for the unknown coefficients $\underline{\underline{a}}$ defined by Equation 3-19.

↓

Substitute the $\underline{a}$ coefficients back into the trial differential equation and solve for the continuous displacement, velocity, and acceleration.

*Optional values

Figure 3-9. Block Diagram for Obtaining a Continuous Response From the Straight-Line Approximation

These specifications are used to generate the straight-line approximations of the state variables of the desired response. The solution of Equation (3-27), subject to the above characteristics, are shown in Figure 3-10. The characteristics of the continuous solution, which fall within 10 per cent of the desired values, are:

(1) Rise time = 5.3 seconds

(2) Per cent overshoot = 6.1 per cent

(3) Settling time = 6.9 seconds (5 per cent settling time)

= 7.3 seconds (3 per cent settling time)

(4) Final value = 0.99.

Three additional straight-line characteristics were assumed in order to generate the straight lines of the approximating method. These characteristics were:

(1) $T_{OS} = 6.5$ Time to maximum overshoot

(2) $T_d = 0.1$ Delay time

(3) $C_8 = 0.5$ Acceleration constant.

These additional values make up a number of characteristics which are required in the straight-line method but are not included in specifying a desired system performance. T_d was set equal to 0.1 in order to limit the maximum velocity. Increasing the value of T_d would result in a greater maximum velocity.

The effects of changing a number of the straight-line parameters are shown in Figures 3-11, 3-12, 3-13, and 3-14. From a study of Figure 3-11, it can be seen that increasing the time to overshoot T_2 has the effect of increasing both the overshoot and the settling time. Decreasing T_2 reduces both the overshoot and settling time. For small changes in T_2 (approximately 15 to 20 per cent), the rise time of the

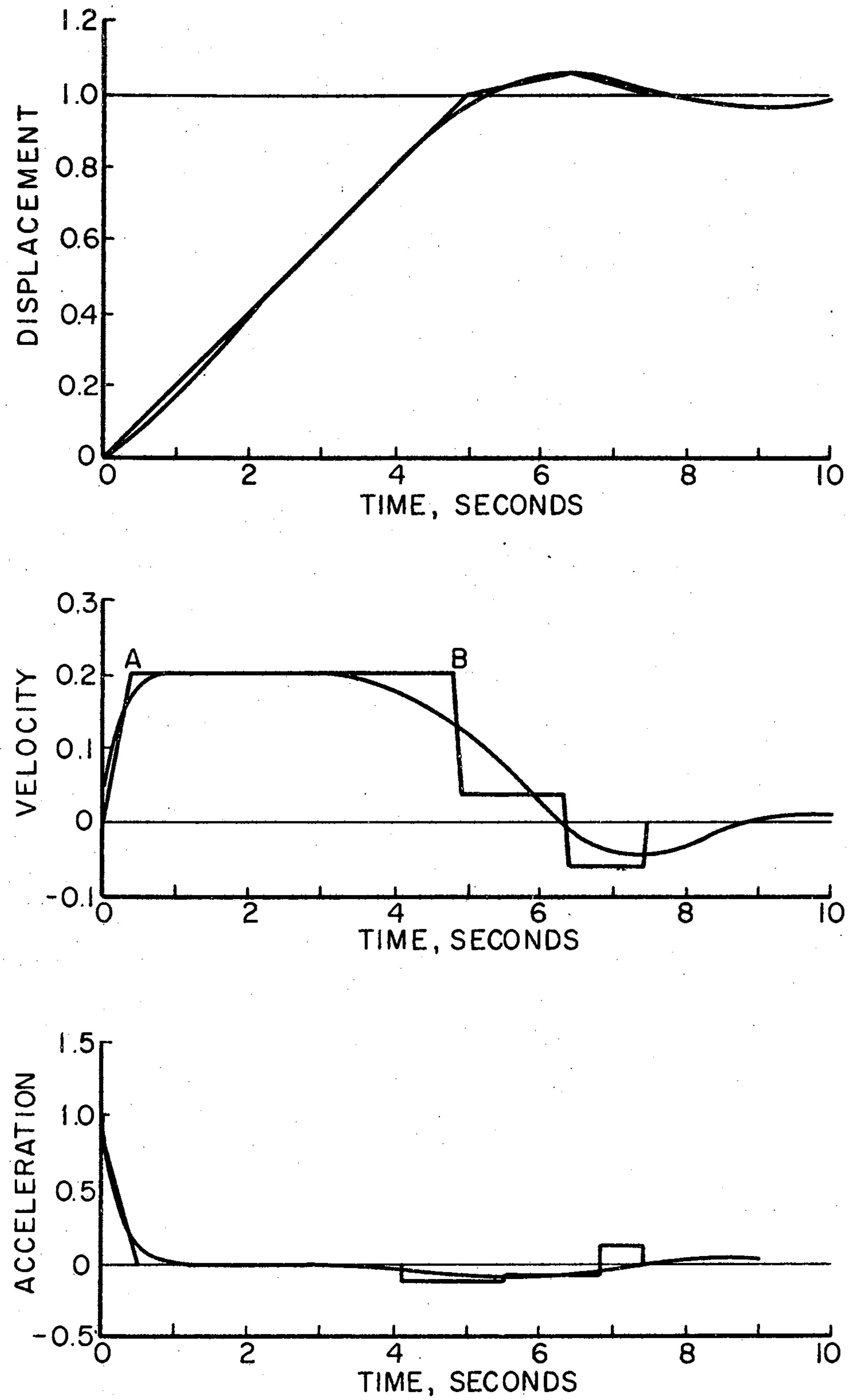
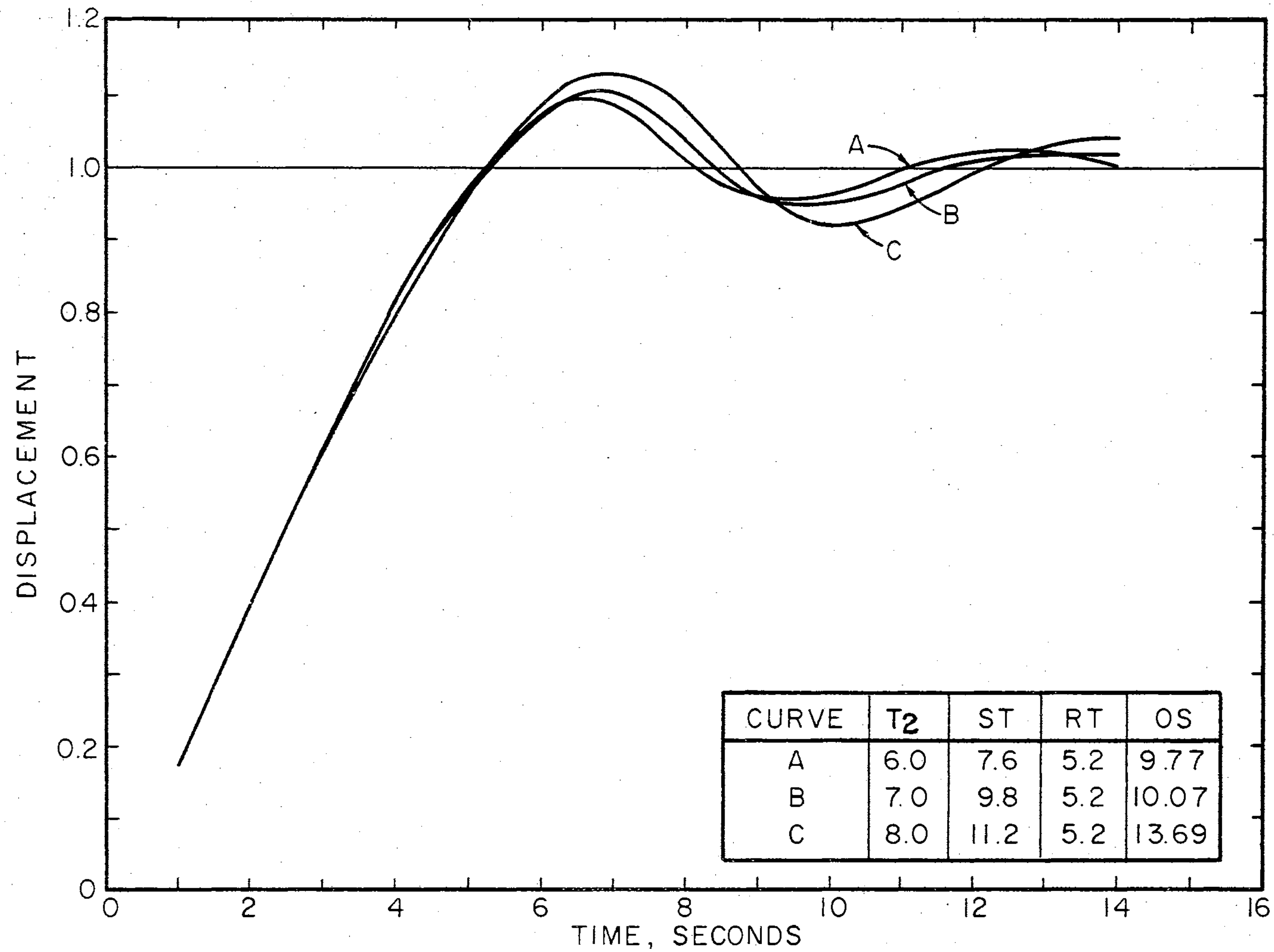
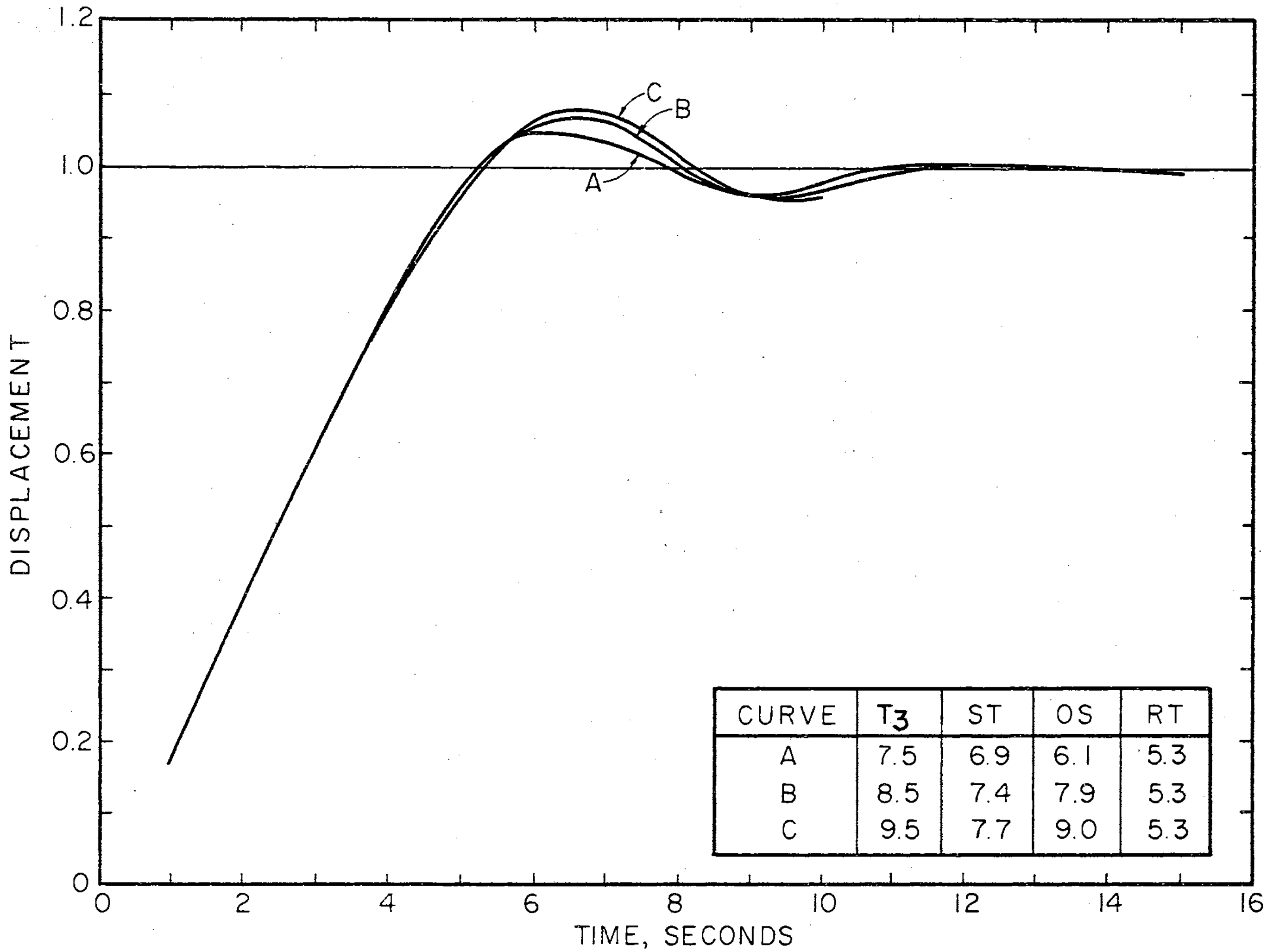


Figure 3-10. Continuous Solution of a Straight-Line Approximation



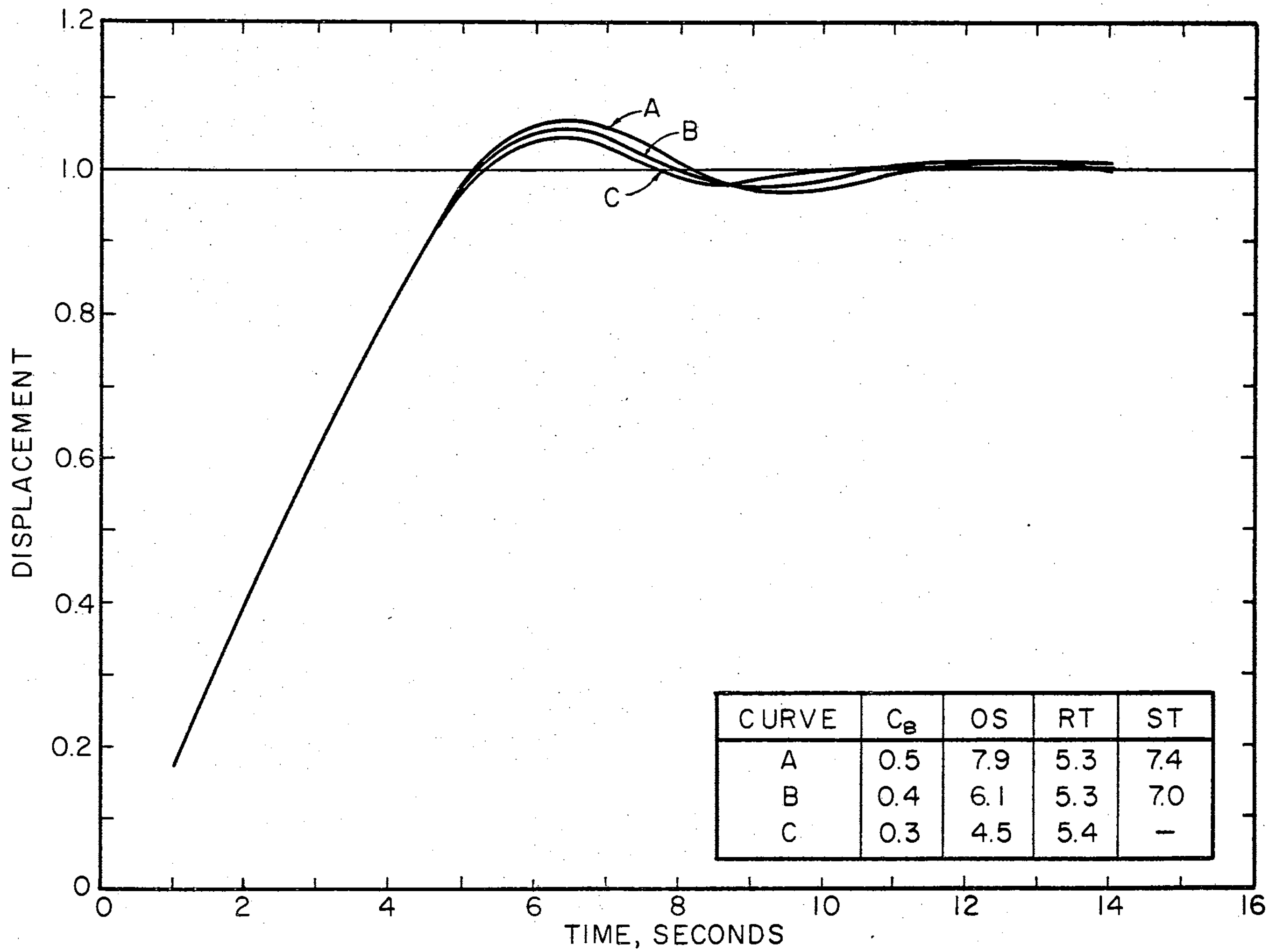
CURVE	T_2	ST	RT	OS
A	6.0	7.6	5.2	9.77
B	7.0	9.8	5.2	10.07
C	8.0	11.2	5.2	13.69

Figure 3-11. Effects of Changing the Time to Overshoot T_2



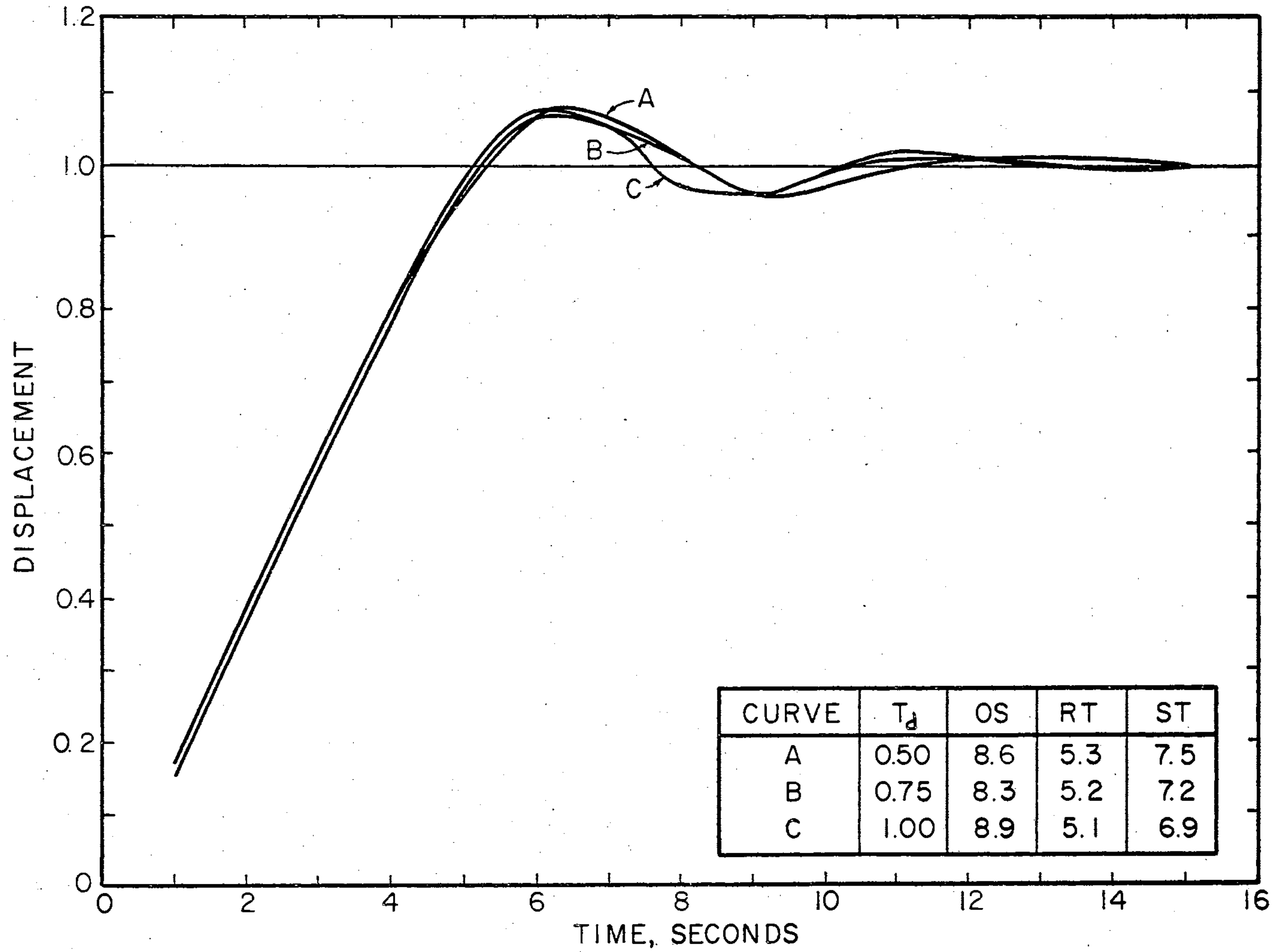
CURVE	T_3	ST	OS	RT
A	7.5	6.9	6.1	5.3
B	8.5	7.4	7.9	5.3
C	9.5	7.7	9.0	5.3

Figure 3-12. Effects of Changing the Settling Time T_3



CURVE	C_8	OS	RT	ST
A	0.5	7.9	5.3	7.4
B	0.4	6.1	5.3	7.0
C	0.3	4.5	5.4	—

Figure 3-13. Effects of Changing the Acceleration Constant C_8



CURVE	T_d	OS	RT	ST
A	0.50	8.6	5.3	7.5
B	0.75	8.3	5.2	7.2
C	1.00	8.9	5.1	6.9

Figure 3-14. Effects of Changing the Delay Time T_d

continuous solution is somewhat insensitive to these changes. The effects of changing the settling time T_3 are shown in Figure 3-12. Increasing the value of T_3 results in an increase in the settling time and overshoot value.

Decreasing the value of the acceleration constant C_8 from 0.5 to 0.3 has the effect of decreasing the maximum overshoot and also the settling time. The effects of this change are shown in Figure 3-13. Changing the delay time T_d has the effect of decreasing the overshoot and the rise time. The effects of this parameter change are shown in Figure 3-14.

The continuous solution of Figure 3-10 follows closely the maximum values set by the straight-line approximation. The velocity curve did not exceed the maximum value, placed on the velocity by the straight line AB, by more than five per cent. The assumed form of the straight-line approximation constrains the velocity and acceleration to meet predetermined maximum values. By assuming a different set of straight-line parameters (delay time, T_d; acceleration constant, C_8; etc.) a different form of system characteristics is obtained. For example, the state variable of a system forced to follow the straight-line approximation shown in Figure 3-15 will exhibit a somewhat different set of characteristics. The maximum velocity of this approximation is 0.23, whereas the maximum velocity of the response given in Figure 3-10 was 0.21.

Assuming a particular form of the straight lines places constraints on the system which may not be met by the particular system; that is, the system may not exhibit the proper rise time when constrained to follow a specific form or maximum velocity. If the system characteristics are not met, two alternatives exist:

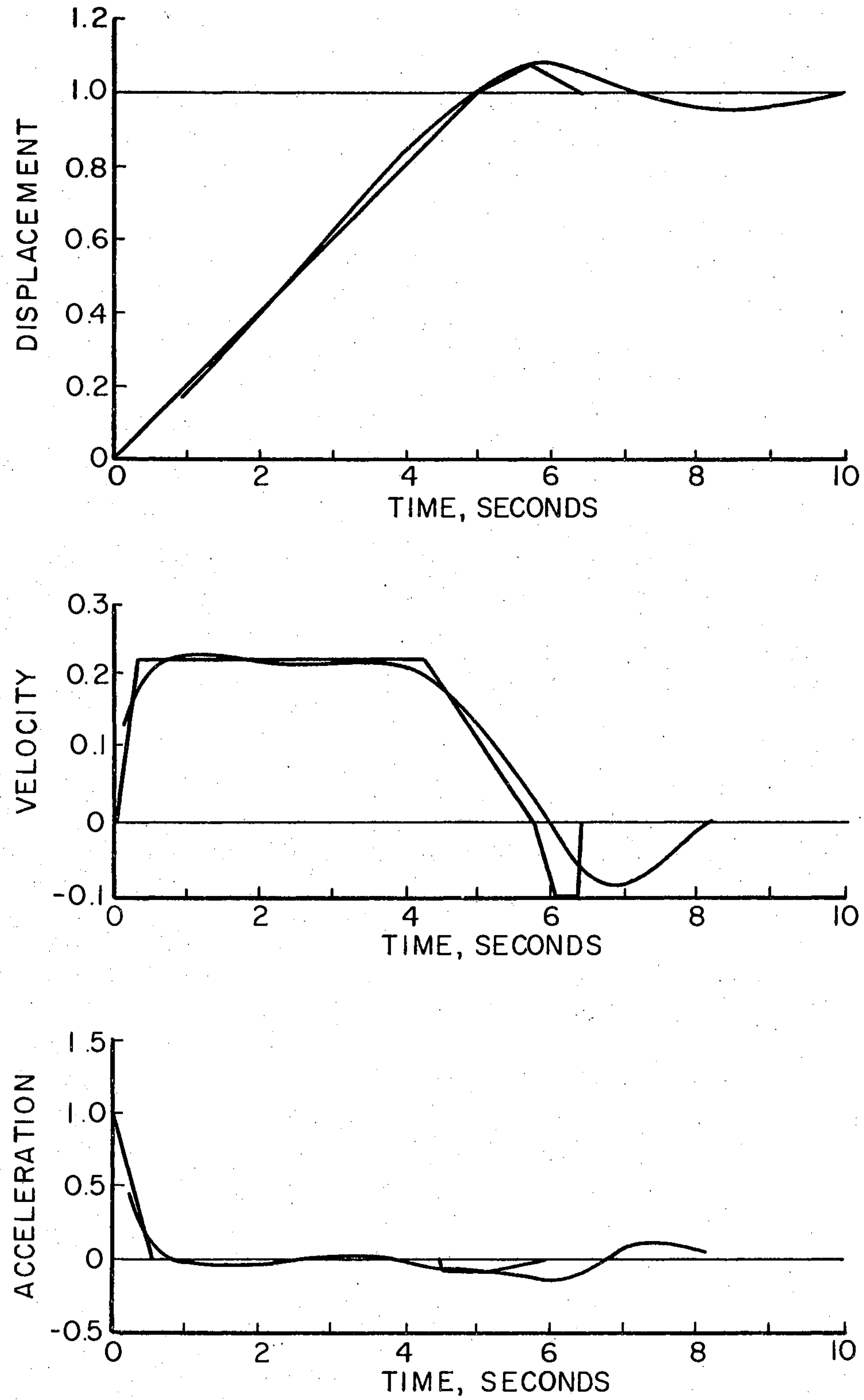


Figure 3-15. Effects of Changing the Form of a Straight-Line Approximation

(1) increase the number of nonlinear terms in the system equation or

(2) change the form of the velocity and acceleration curves by changing the parameters of the straight-line approximation.

The choice of which method is to be used depends on the design constraints. The first has the effect of complicating the physical design whereas the second method may result in excessive values of the state variables. The changing of the form of the system parameters by adjusting the straight-line approximation is presented in Chapter V as a refinement of the method.

Hove (11) showed that increasing the number of nonlinear terms used in the fitting method resulted in a better fit but a somewhat complicated design. The physical realizability of these elements limits the form of the nonlinearity which can be used in the analysis procedure. A discussion of a set of nonlinear elements which are defined for a fluid-mechanical system is presented in the next section.

Control Parameter Determination

The dynamic system which is to be controlled is characterized by a set of differential equations of the form

$$\frac{dX_i}{dt} = f_i(X_1, X_2, \ldots, X_N) \tag{3-7}$$

To this system equation is added a control vector g_i which is to force or cause the system to perform in a specified manner. This vector control function has been previously defined in Equation (3-9) as

$$g_i = g_i(a_{il}u_{il})$$

$$l = 1, 2, \ldots, q_i. \tag{3-9}$$

The form as well as the magnitude of the terms of the uncontrolled system is known and is determined by mathematically modeling the uncontrolled physical system. The magnitude of the control vector g_i is unknown and is to be determined in the analysis procedure. Before the analysis procedure may be carried out, the form of the control vector must be established.

The terms in the control vector should be made up of a set of physically realizable control parameters which are defined for a specific system. Given the form of the control vector, the analysis method involves determining the magnitude of the unknown coefficients a_{il}, $l = 1, 2, \ldots, q_i$ defined by Equation (3-9).

The form of the terms which make up the control vector g_i are determined by mathematically modeling the system elements which may be altered or introduced into the system. An example for determining the control parameters for a hydraulic spool valve is given in the next section.

Example: Application of the Analysis Method to a Hydraulic Spool Valve

The analysis of the hydraulic spool valve is a conventional presentation of considerable value for demonstrating the application of the analysis method. The analysis method consists of determining the mathematical model of the fixed system and then choosing a control vector which is to be added to the system to bring about a desired system response. The hydraulic spool valve discussed in this section is shown in Figure 3-16. The differential equation which describes the motion or response of this system to a unit step function is found by determining the applicable axial forces which act upon the valve spool. The forces which act upon the spool can be separated into two distinct classes; that is, the mechanical forces and the fluid forces. The fluid forces which exist within a defined control volume V_A of the spool valve may be expressed mathematically as (5)

$$F_x = \frac{d}{dt}\int_{V_A} \rho V_x dV_A + \int_{A_S} \rho V_x V_N dA_S \tag{3-28}$$

where

V_A = control volume of fluid bounded by the spool and the housing (see Figure 3-16)

ρ = fluid density

V_x = x-component of the fluid velocity

V_N = component of the fluid velocity normal to the control volume surface.

A_S = surface area of the control volume.

This mathematical expression defines the reaction forces on the spool due to the time rate of change of fluid momentum within the system. The two integrals on the right hand side of Equation (3-28) are commonly referred to as the unsteady and steady flow force terms, respectively. The force generated as a result of the steady flow component is assumed to be only a function of the displacement. This requires that the angle θ and the pressure drop $(P_s - P_e)$ are constant for all values of the displacement X. The steady flow force component is described mathematically as

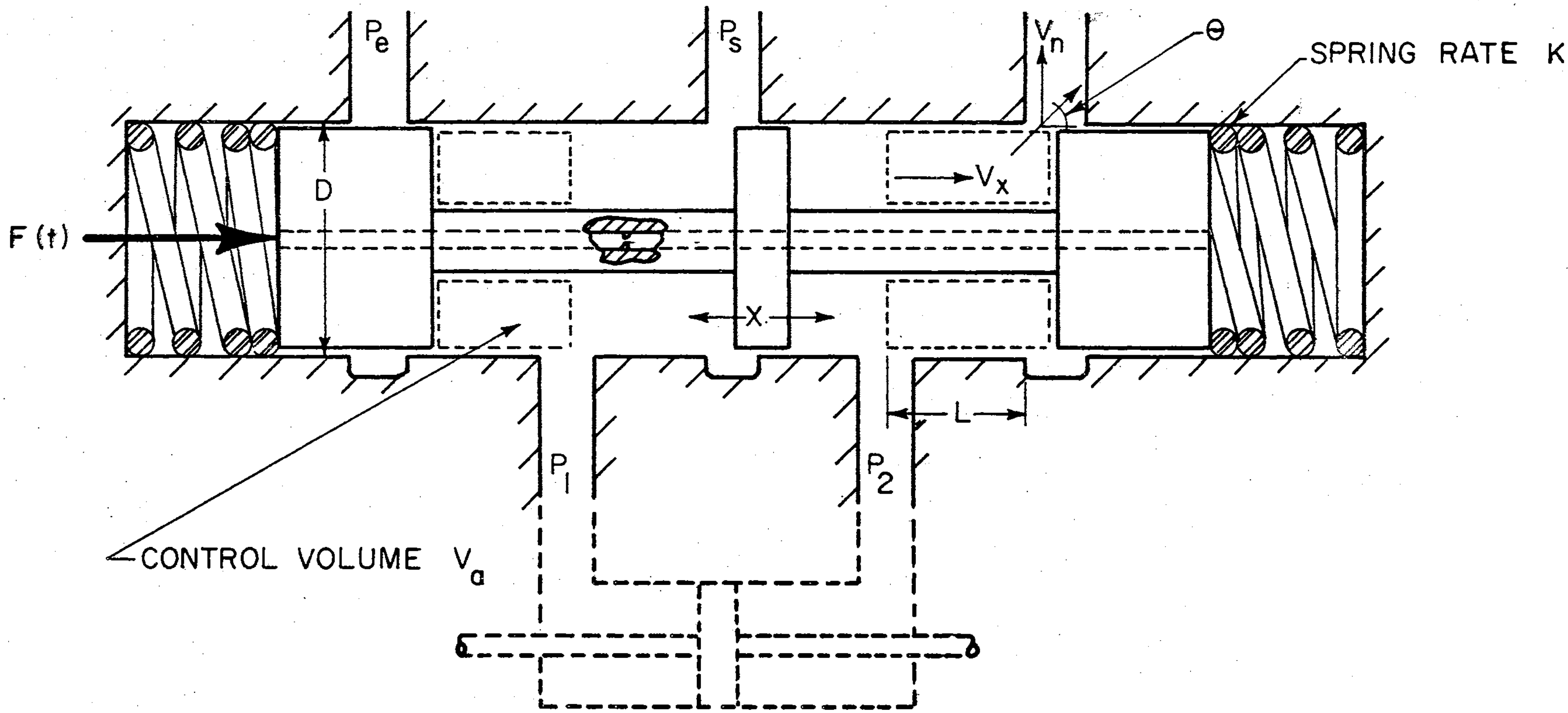


Figure 3-16. Hydraulic Spool Valve

$$(F_x)_s = A_1 X \tag{3-29}$$

where

$A_1 = 2C_d w \cos\theta(P_s - P_e)$

X = valve stem displacement

P_s = supply pressure

P_e = tank pressure

w = peripherial width of the spool

θ = angle at which the fluid enters the control volume.

The unsteady flow force $(F_x)_{us}$, defined by

$$(F_x)_{us} = \frac{d}{dt}\int_{V_A} \rho V_s dV_A \tag{3-30}$$

is a result of fluid acceleration induced by pressure changes and/or valve displacement. For the control volume, V_A (see Figure 3-16), the unsteady flow force $(F_x)_{us}$ can be shown to be a nonlinear function of the displacement and velocity (15)

$$(F_x)_{us} = A_2(L + X)\dot{X} \tag{3-31}$$

where

$A_2 = C_d w\sqrt{\rho(P_s - P_e)}$

L = characteristic length of the control volume

$\dot{X}$ = valve spool velocity.

The viscous drag force is defined by

$$F_D = A_3\dot{X} \tag{3-32}$$

where

$A_3 = \mu\pi D\ell/y$

μ = fluid viscosity

D = spool diameter

ℓ = contact length of spool and housing

y = radial clearance.

The force due to compression of the spring is modeled by

$$F_{sp} = KX \tag{3-33}$$

where K is the spring rate. These forces are summed into differential equation form by application of Newton's second law

$$(M + \frac{M_s}{3})\ddot{X} + [A_3 + A_2(L + X)]\dot{X} + (K + A_1)X = F(t) \tag{3-34}$$

where

M_s = mass of the spring

M = mass of the spool

F(t) = forcing function.

In the preliminary design stages, the characteristic length of the control volume (L + X) is taken to be

$$L_1 = L + \tfrac{1}{2}(X)_{max} \tag{3-35}$$

where $(X)_{max}$ is the maximum spool displacement. Combining Equations (3-34) and (3-35) yields

$$\ddot{X} + B_1\dot{X} + B_2X = F(t). \tag{3-36}$$

Assume that the coefficients B_1 and B_2 have values of 0.36 and 0.24, respectively. The spool valve modeled by

$$\ddot{X} + 0.36\dot{X} + 0.24X = 0.24 \tag{3-37}$$

has an overshoot of 29 per cent and a rise time of 4.3 seconds.

In order to improve the dynamic characteristics, a control vector g_i must be defined for the spool valve. The elements of the control vector must be physically relizable and capable of forcing the dynamic response to behave in a predescribed manner. The control vector g_i for this example will consist of two nonlinear elements. The first element is a "hard" spring modeled by $F_{sp} = C_1X^3$ and the second is the unsteady flow component $C_2X\dot{X}$. With these two additions, the hydraulic valve is modeled by

$$\ddot{X} + 0.36\dot{X} + 0.24X + C_1X^3 + C_2X\dot{X} = 1 \qquad (3\text{-}38)$$

where C_1 and C_2 are the coefficients of the predetermined control vector g_i which are to be determined by the analyses method. For example, if the specification for the above example required a rise time of 2.0 seconds and an overshoot value of 10 per cent, then the system

$$\ddot{X} + 0.36\dot{X} + 0.24X + 0.801X^3 + 1.4039X\dot{X} = 1 \qquad (3\text{-}39)$$

would come within five per cent of these requirements. The mechanics of the analyses method for obtaining the coefficients C_1 and C_2 are detailed in Chapter V.

The two control elements used in this example are used only to demonstrate the analysis method. There are a number of additional control elements which can be added to the spool valve to bring about a desired performance. For example, flow through the sharp edge orifice shown in Figure (3-16) will produce a "square law" damping force of the form

$$(F_d)_{orifice} = D_o\, v/v/ \qquad (3\text{-}40)$$

where D_o is defined as the orifice coefficient and v is the velocity of flow through the orifice. The addition of "hard" and "soft" springs as control elements can also be used. The form of the control elements using nonlinear springs is

$$F_{sp} = f(X). \tag{3-41}$$

These nonlinear elements are of the type that can be incorporated into the control vector g_i in order to bring about a desired performance. The particular choice of the nonlinear elements which should be used is dependent on the specific problem and its application.

CHAPTER IV

ANALOG COMPUTER STUDIES

The theoretical analysis and techniques presented in Chapter III are applicable to stationary (constant coefficients) second-order nonlinear differential equations. The choice of the type of system to be analyzed is restricted only to this class of system.

The physical system used in this study was an electrical network whose mathematical model is a second-order nonlinear differential equation. This choice of circuit was chosen since the available components for the electrical network were within 1 per cent of their rated value.

The electrical network was implemented on an Electronic Analog Computer (EAI TR-48). The solution of this network on the analog computer will be analogous to a fluid-mechanical system considered in the example of the previous section since the dynamic equations are identical. The Electronic Analog Computer provides a continuous accurate solution of the differential equations describing a physical system with the added advantage of easily changing the physical characteristics or parameters of the system.

Experimental Procedure

The particular system equation studied was a second-order nonlinear differential equation of the form

$$\ddot{X} + A_1\dot{X} + A_2X + A_3X^3 + A_4X\dot{X} + A_5X^2 = 1. \qquad (4\text{-}1)$$

The experimental study was used to check the results of the digital computer solution and to check the sensitivity of the coefficients of the solution to small parameter changes. The circuit diagram for the nonlinear equation is shown in standard block diagram form in Figure 4-1.

The procedure for experimentally checking the solution method was as follows:

(1) A set of system coefficients of the continuous solutions was programmed on the analog computer by setting the appropriate potentiometers.

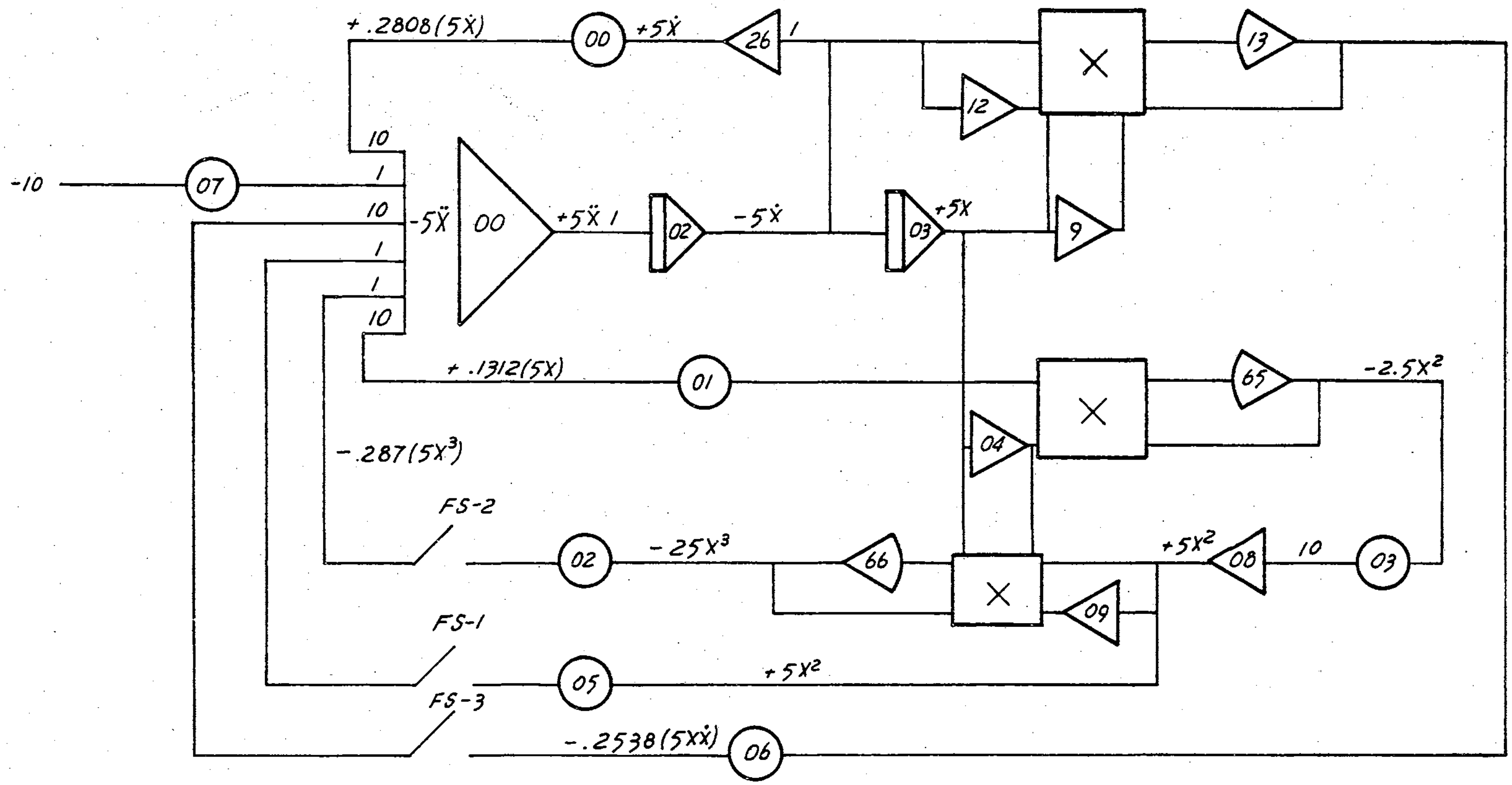
(2) The physical system was then subjected to a unit step input.

(3) The output of the system was recorded on a T-Y plotter (EAI Model 1110).

(4) Steps 1 through 3 were then repeated for several sets of coefficients.

Several different sets of values of the system equation were used to check the analysis method. Some of the results of the method are given in Figures 4-2 and 4-3. Figures 4-2 and 4-3 show the analog results of two fitted equations. The straight-line characteristics of the numerical solution are shown on the figures.

The measured response (plotted with a T-Y plotter) shown in Figure 4-2 shows that the electrical circuit was controlled within the tolerance range of the numerical procedure. The required rise time of 5.6 seconds and overshoot value of six per cent were within



$$\ddot{X} + 2.808\dot{X} + 1.312X - .287X^3 - 2.538X\dot{X} = 1$$

SCALED by 5

SCALED EQUATION

$$-5\ddot{X} = +2.808(5\dot{X}) + 1.312(5X) - .287(5X^3) - 2.538(5X\dot{X}) - 5$$

Figure 4-1. Block Diagram of a Second-Order Nonlinear Differential Equation

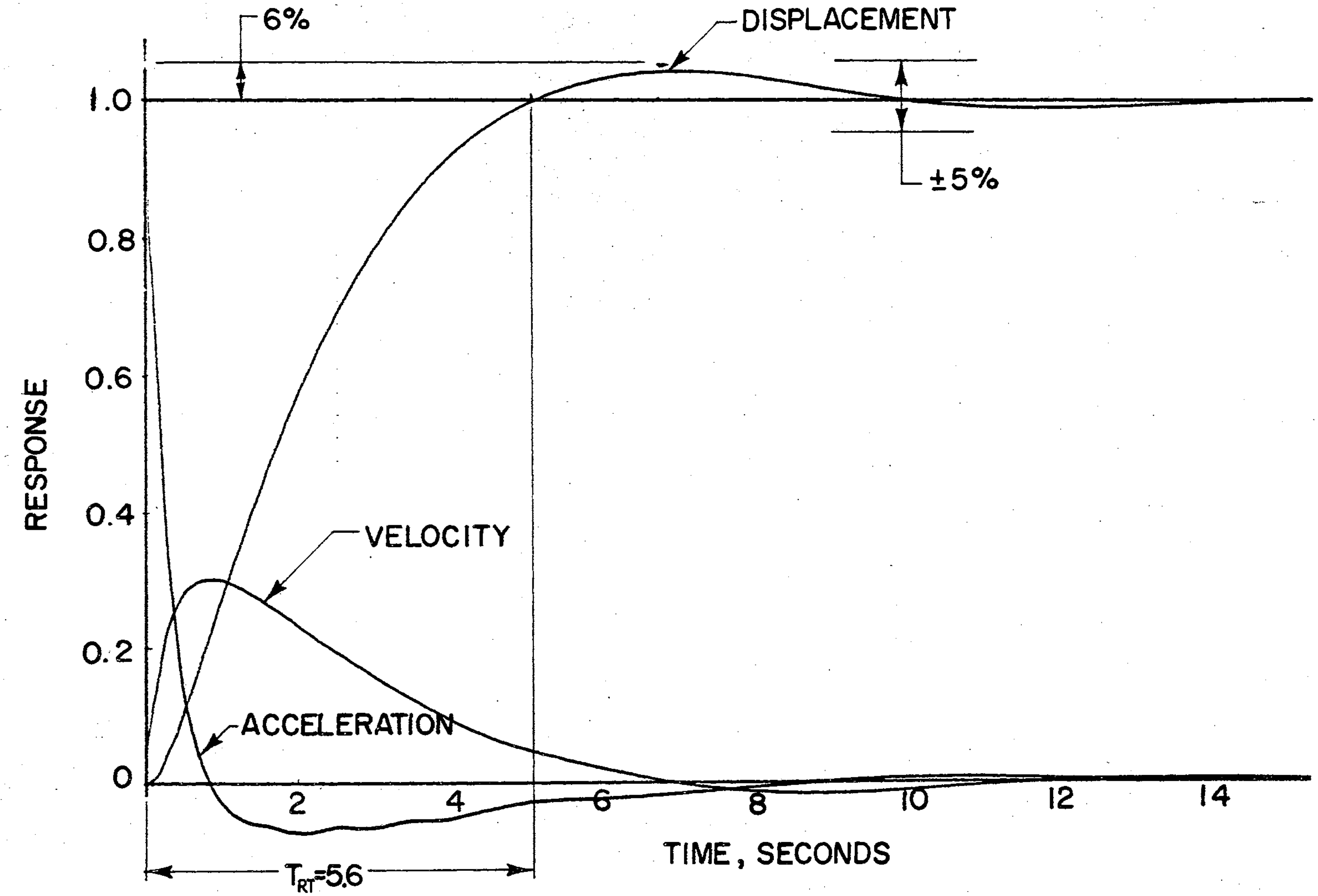


Figure 4-2. Analog Computer Results $T_{RT} = 5.6$

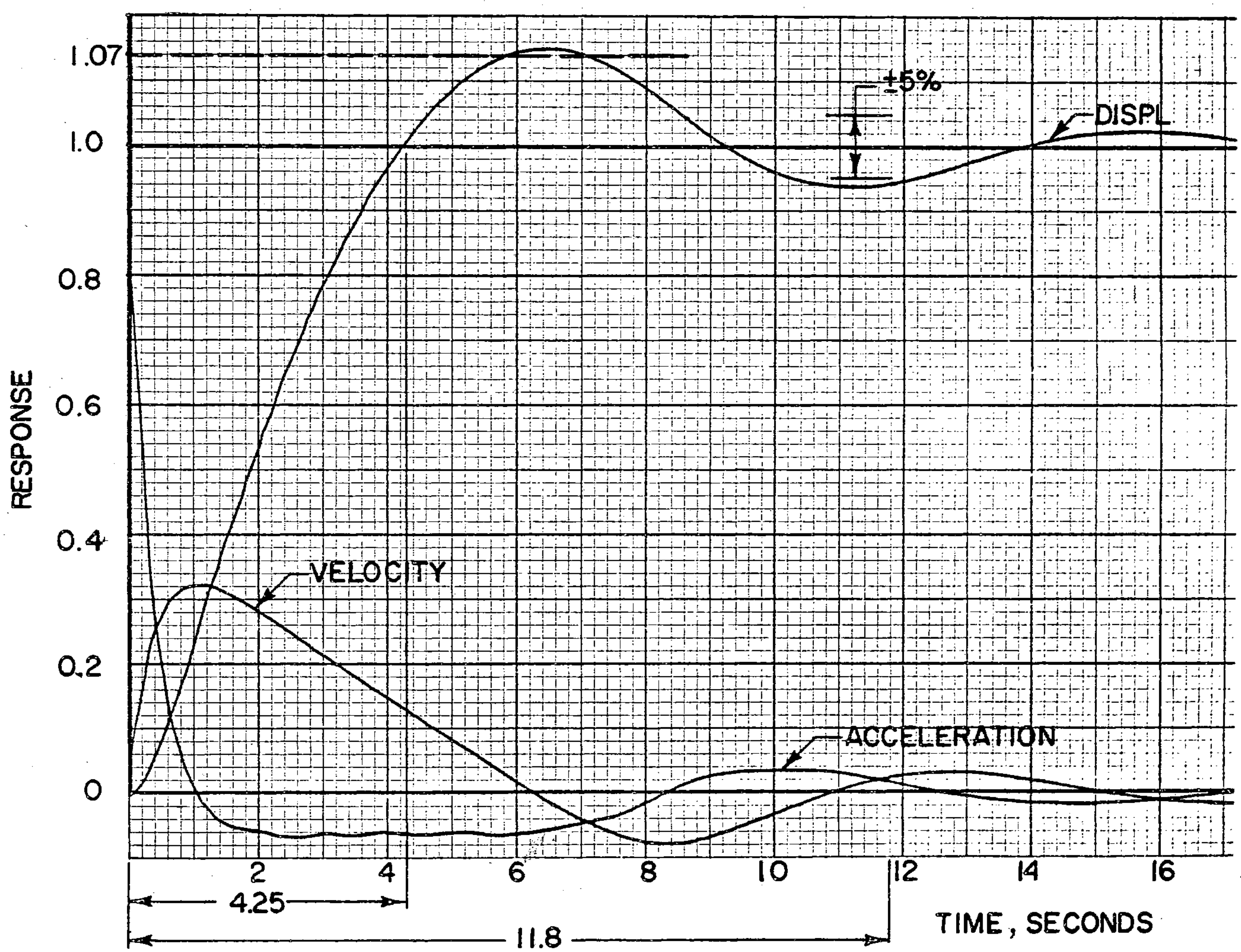


Figure 4-3. Analog Computer Results $T_{RT} = 4.25$

5 per cent of the predicted values. The response shown in Figure 4-3 falls within the 5 per cent tolerance range as specified by the numerical procedure. The differences between the numerical procedure and the measured response may be attributed to errors in the recording equipment and the rated values of the physical components. The agreement between the numerical procedure and the electrical circuit was quite good when considering the variation in the physical components of the electrical circuit.

Coefficient Sensitivity

The coefficients of a mathematical model are functions of the physical characteristics of the system. These characteristics (fluid temperature, fluid volume, spring rates, etc.) of the system may change during operation. The mathematical model theoretically should take care of any of the changes which occur. However, in the derivation of the system differential equation, certain assumptions are made which may be entirely valid; but small and large pertubations about these values may radically affect the desired response.

The analog computer offers a convenient and rapid means of determining the effects of small parameter changes. This change in the system parameter is accomplished by varying the coefficients of the system differential equation. Variation of the coefficients by changing the appropriate potentiometer setting will have the same effect as changing the system parameters, such as valve spool size, spring rate, and nonlinear damping. The effects of changing these parameters of the system were studied by using the analog computer.

The results of the study showed that the desired response was not largely affected by small changes in the nonlinear terms involving the displacement. The effects on the system performance were quite significant when the coefficient of the nonlinear term $X\dot{X}$ was perturbed about its fitted value. For the differential equation

$$\ddot{X} + 2.808\dot{X} + 1.312X - 0.287X^3 - 2.538X\dot{X} = 1. \qquad (4\text{-}2)$$

An increase in the coefficient of the $X\dot{X}$ term of about 1 per cent resulted in a 6 per cent change in the overshoot value. A check of this solution on the digital computer showed that changing this value from -2.538 to -2.6649 resulted in an unstable system. The question of using this design arises if the value of the coefficient is likely to change during system operation. A system defined by the equation

$$\ddot{X} + 4.694\dot{X} + 0.7507X + 0.2486X^3 - 3.862X\dot{X} = 1 \qquad (4\text{-}3)$$

does not exhibit these same characteristics for small changes of the $X\dot{X}$ term. A change in this coefficient from -3.862 to -4.081 resulted in only slight changes in the output response (overshoot increase from 6.1 to 8.9 per cent and rise time decrease from 5.3 to 5.0 seconds). A more detailed analysis of the sensitivity of the various nonlinear terms is not necessary to check the validity of the thesis method.

Summary of Test Results

The experimental results show very clearly (within the accuracy of the analog computer and recording equipment) that the least squares method of fitting a given set of system characteristics can be made with a high degree of accuracy. The accuracy of the solutions shown in Figures 4-2 and 4-3 are within 5 per cent of the numerical procedure.

The choice of a preliminary design should certainly include an analysis of the sensitivity of the terms with respect to small changes in their magnitude before it is included as an acceptable control parameter.

CHAPTER V

REFINEMENT OF THE ANALYSIS METHOD

The analysis method of Chapter III presents a method whereby a second-order nonlinear differential equation is fitted to a set of desired system characteristics. This fit is accomplished by transforming the desired characteristics into a straight-line approximation of the state variables of the system. These straight lines are then fitted to a second-order nonlinear differential equation by a least squares procedure. This differential equation is then solved by a Runge-Kutta solution to obtain a continuous solution of the state variables of the system.

The straight-line approximation developed in Chapter III places constraints on the form of the velocity and acceleration curves as well as satisfying the desired system characteristics (rise time, settling time, and overshoot). In many cases the designer is not concerned with the velocity and/or acceleration requirements but only with a solution which satisfies the desired system characteristics. In this chapter, an iterative solution using the straight-line method of Chapter III, which will force the dynamic response to come within a specified tolerance, is presented. The iterative procedure of this thesis does not place constraints on the velocity or acceleration but only on satisfying the desired system characteristics of rise time, overshoot, and settling time.

However, the methods developed in this thesis are not restricted to such limiting constraints.

Iterative Solution

The results of the parameter study of the straight-line approximation method shown graphically in Figures 3-11 through 3-14 are used to establish the iterative procedure. The straight-line approximation of the desired system characteristics is the basis from which all changes in the iteration procedure are made. The results of this analysis indicated definite trends so that a logical decision could be made as to what changes would result in a more desirable fit.

From a study of Figures 3-11 through 3-14, it was observed that changing the overshoot and settling time values of the approximation had little effect on rise time. Therefore, the value T_1 is adjusted first. The magnitude of the change is a function of the error between the desired value and the fitted value. For example, if the desired rise time is $t = 5$ seconds, and the value of the continuous response at $t = 5$ seconds is 0.97, then a rise time correction of

$$(T_1)_{adjusted} = T_1 + (T_{RT} - T_{fitted}) \qquad (5\text{-}1)$$

is used. The straight-line approximation is then recalculated and the rise time value rechecked. The procedure continues until the rise time comes within a specified tolerance value.

The next step is to examine the overshoot and settling time values. If both fail to meet the desired specification, the values T_2 and T_3 are adjusted by a proportionality constant K_1 which is equal to

$$K_1 = 1 - (\text{Fitted Value} - \text{Desired Value}) \qquad (5\text{-}2)$$

For example, if the desired overshoot is 10 per cent and the fitted value is 9 per cent, a correction of the time to overshoot value of

$$T_3 = T_3(K_1) = T_3(1.1) \tag{5-3}$$

is made.

The logic of the complete iterative procedure is given in block diagram form in Figure 5-1. A Fortran digital program which performs the logic and makes the changes is given in Appendix C.

Results of the Iteration Method

The results of the iterative procedure on a number of different types of systems and response characteristics are presented. The following specification for a desired system performance was assumed:

$$T_{RT} = 3.5 \text{ seconds}$$
$$T_{ST} = 5.5 \text{ seconds}$$
$$OS = 1.10$$

Several values associated with the straight-line approximation were assumed (see Appendix C) in the iterative procedure. They are:

$$\begin{aligned} T_{OS} &= T_{RT} + \tfrac{1}{2}(T_{ST} - T_{RT}) \\ T_d &= 0.1\ (T_{RT}) \\ C_8 &= 0.5 \end{aligned} \tag{5-4}$$

These values are defined in Figures 3-4 and 3-6.

The dynamic system which was to be fitted to these specifications is given by

$$\ddot{X} + A_1\dot{X} + A_2X + A_3X^3 + A_4X\dot{X} = 1 \tag{5-5}$$

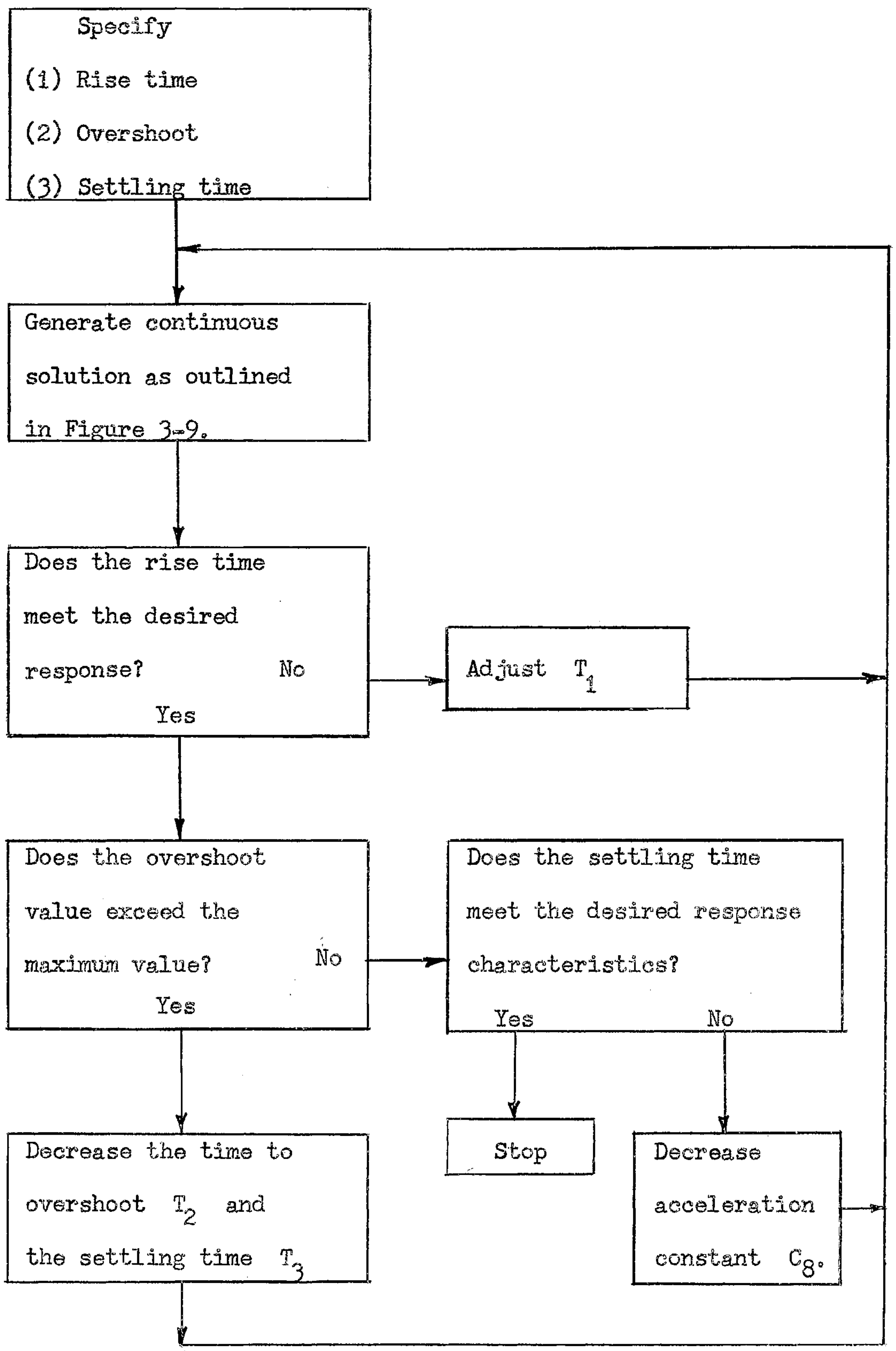


Figure 5-1. Block Diagram for Iterative Procedure

The values of the system characteristics obtained during the iteration procedure are given in Table II and the results are shown graphically in Figure 5-2.

TABLE II

RESULTS OF THE ITERATIVE PROCEDURE

Iteration Number	Rise Time	Settling Time	Overshoot Per cent
-*	3.5	5.50	10.00
0**	3.9	5.50	7.49
1	3.5	5.40	8.39
2	3.5	5.40	8.34
3	3.5	5.50	8.40
4	3.5	5.60	9.36
5	3.5	5.70	9.84
6	3.5	5.75	9.98

*Desired Response
**Solution of the unadjusted straight-line approximation

The results of the iterative procedure agree within five per cent (the tolerance value placed on the system characteristics) of the desired values. A better fit is possible if a closer tolerance range is specified. Three other sets of system characteristics for the dynamic system represented by Equation (5-5) were fitted with the iterative procedure. The fitted responses are shown in Figure 5-3.

Given the continuous solution of a desired performance specification, it is then possible to determine the optimal value (in a least squares sense) of the coefficients of the nonlinear terms

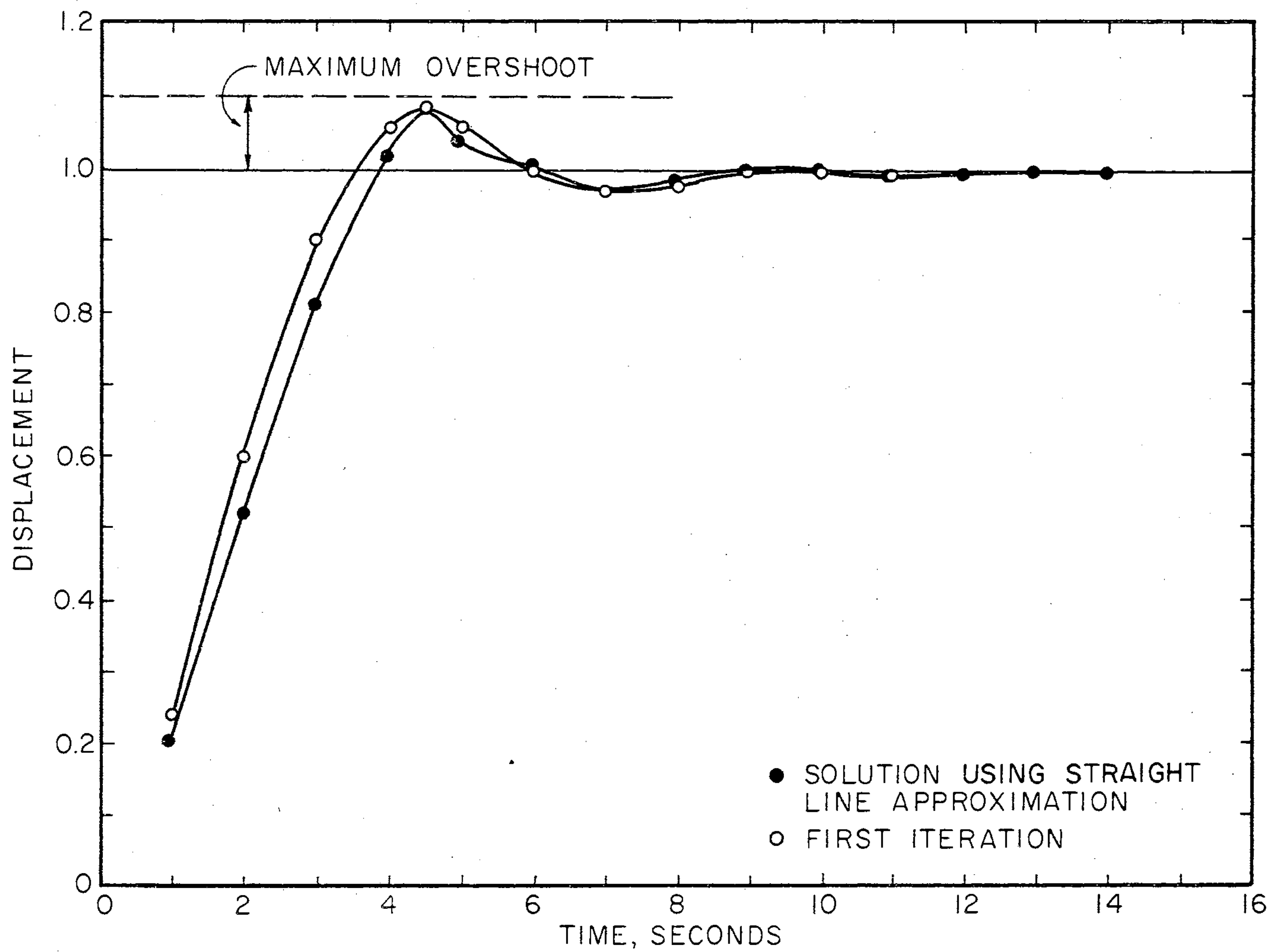


Figure 5-2. Continuous Solutions of Iterative Procedure

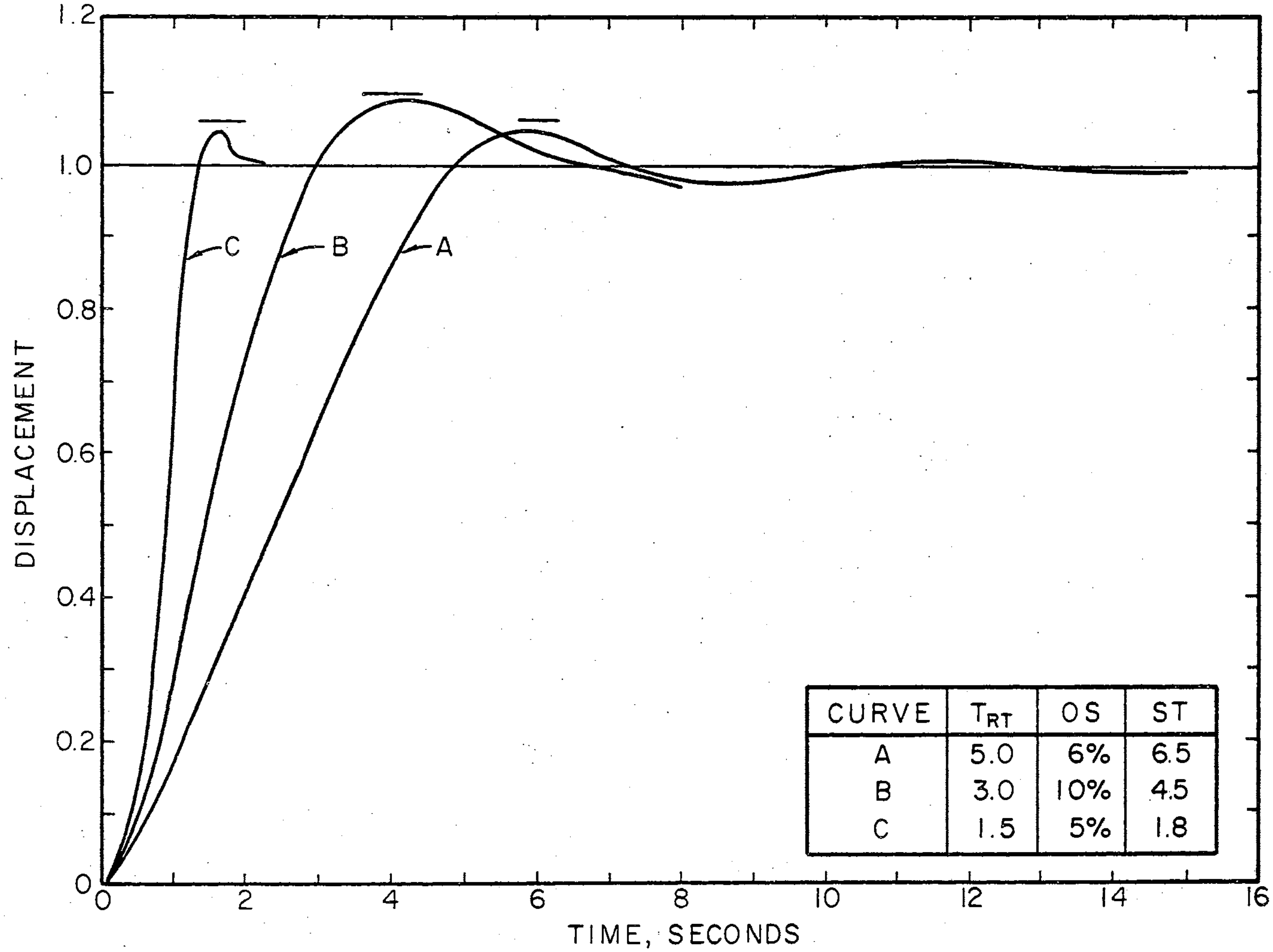


Figure 5-3. Results of the Iterative Solution

which make up the control vector g_i. If some of the coefficients of the system are fixed, then the normal equations of the nonlinear differential equation take a different form. An example is presented in the next section to illustrate the fitting of a dynamic system with two fixed coefficients to a set of system characteristics.

Application of the Straight-Line Method to a Dynamic System with Two Fixed Coefficients

The procedure for applying the method to a system with some fixed coefficients is presented in this section. The analysis method requires that the normal equations be obtained. The assumed equation form is

$$\ddot{X} + B\dot{X} + KX + A_3X^3 + A_4X\dot{X} = 1 \tag{5-6}$$

The two coefficients B and K are of fixed value while A_3 and A_4 are variable and are to be used to control the response.

The procedure for fitting this system to a desired response is as follows:

(1) Reduce the system equation (Equation 5-6) to its state variable representation by making the following substitutions:

$$X = X_1 \tag{5-7}$$

$$\dot{X}_1 = X_2 \tag{5-8}$$

$$\dot{X}_2 = -BX_2 - KX_1 - A_3X_1^3 - A_4X_1X_2 + 1. \tag{5-9}$$

(2) Equation (5-9) is rewritten as

$$\dot{X}_2 + BX_2 + KX_1 - 1 = -A_3X_1^3 - A_4X_1X_2 \tag{5-10}$$

The left-hand side of Equation (5-10) is the fixed part, and the right-hand side is variable since A_3 and A_4 can be changed.

The analysis method consists of selecting m points in time from the state variable representation of the system (the desired response) and then solving this set of m equations by the least squares method. The vector form of the given system is

$$\underset{mx1}{Y} = \underset{mx2}{u} \; \underset{2x1}{A} + \underset{mx1}{e}. \tag{5-11}$$

The terms of this equation are defined by Equations (3-10) through (3-16).

(3) The matrix representation of the normal equation is defined by Equation (3-21). The matrix representation for the system defined by Equation (5-6) is

$$\begin{bmatrix} \Sigma X_1^6 & \Sigma X_1^4 \\ \Sigma X_1^4 X_2 & \Sigma X_1^2 X_2^2 \end{bmatrix} \begin{bmatrix} A_3 \\ A_4 \end{bmatrix} = \begin{bmatrix} \Sigma X_1^3(\dot{X}_2 + BX_2 + CX_1 - 1) \\ \Sigma X_1 X_2(\dot{X}_2 + BX_2 + CX_1 - 1) \end{bmatrix} \tag{5-7}$$

where the summation is taken over m points.

(4) The straight-line representation of the state variables X_1, X_2, and $\dot{X}_2$ is used as the input data to the system.

(5) The desired system characteristics are then used to generate the state variable approximation.

(6) The solution is then checked with the desired response. If the solution does not agree, then the iteration procedure developed in the previous section is used to improve the fit.

These steps outline the procedures for fitting a system with fixed coefficients to a desired system response. The particular example is for two fixed values, but the method is easily extended for other combinations.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

The design of nonlinear second-order dynamic systems to meet arbitrary performance criteria has been investigated with particular reference to fluid-mechanical systems. This thesis concentrates on development of a method for designing the system to meet specific system requirements. The method, given a set of desired system characteristics (rise time, overshoot, and settling time), requires construction of a straight-line approximation of the state variables of the system. The approximation is used to generate a continuous set of state variables which satisfy the system requirements. An iterative procedure is presented which forces the dynamic response to come within a specified tolerance.

The analysis method offers the following advantages over current methods:

(1) Given a set of desired system characteristics, a continuous representation of the state variables of the system can be generated.

(2) Constraints can be placed on the velocity and acceleration variables by placing restrictions on the form of the straight-line approximation.

(3) The "goodness" of the system fit to the desired characteristics is not a function of the number of

selected points. This is a result of the state variables being obtained from the solution of a nonlinear differential equation.

(4) A number of different forms of the velocity and acceleration curve can be studied with the same desired system characteristics and equation form. This is done in order to obtain a best fit with a limited number of nonlinear terms.

(5) The effects of specific types of compensator forms can be studied.

There are a number of natural extensions of the work presented in this investigation. The following recommendations for further study are made:

(1) The methods used in this study should be extended to include higher order systems. The study should include an approach to component coupling.

(2) An investigation should be made to see if the analysis method could be used to update present mathematical modeling techniques. This would require an experimental testing program to obtain the state variables description of the system.

(3) The use of the method for system synthesis should be investigated.

(4) A study should be made of the stability of the solutions which are determined by the analysis procedure.

(5) A detailed study should be made of the actual dynamics of some of the more common compensating devices, such as the sharp edge orifice, unsteady flow force model, etc.

(6) A study should be made of other forms of straight-line approximation methods. The method presented in this thesis forces the system to fall within the maximum values set by the straight-line method. A different form of approximation may allow a better control with fewer nonlinearities.

(7) The analysis method uses an iterative process which adjusts the parameters of the approximating method. How well these adjustments or corrections are made determines how quickly the solution converges to the final desired value. A study should be made to determine a set of correction factors which would cause the solution to converge more quickly.

(8) The method of analysis presented in this thesis is restricted to the design of a system with a specific step input. A study should be made to see if the analysis method could be used to design a system which would exhibit the same set of dynamic characteristics for a range of step values.

SELECTED BIBLIOGRAPHY

(1) Ayres, M. U. "A Theoretical and Experimental Comparison of Matrix Methods for Structural Analysis." (unpublished Doctor's thesis, Oklahoma State University, 1966).

(2) Bellman, Richard. Adaptive Control Processes: A Guided Tour. Princeton: Princeton University Press, 1961.

(3) Bellman, R., H. Kagiwada, and R. Kalaba. "A Computational Procedure for Optimal System Design and Utilization." Proc. National Academy of Science. Vol. 48. (1962) 1524-1528.

(4) Bernhart, Walter D. "Determination of Parameters and Operations of Dynamic Systems." (unpublished Doctor's thesis, Oklahoma State University, 1964).

(5) Blackburn, J. F., G. Reethof, and J. L. Shearer. Fluid Power Control. Cambridge: The M.I.T. Press, 1960.

(6) Bose, Robert E. "The Effect of Cavitation on Particulate Contamination Generation." (unpublished Doctor's thesis, Oklahoma State University, 1966).

(7) Chen, Kan. "Quasi-Linearization Design of Nonlinear Feedback Control Systems." IEEE Transactions on Automatic Control. Vol. AC-9. (May, 1964), 189-194.

(8) D'Azzo, John J., and Constantine Houpis. Feedback Control System Analysis and Synthesis. New York: McGraw-Hill Book Company, Inc., 1960.

(9) Gibson, John E. Nonlinear Automatic Control. New York: McGraw-Hill Book Company, Inc., 1963.

(10) Gille, J. C., M. J. Pélegrin, and P. Decauline. Feedback Control Systems. New York: McGraw-Hill Book Company, Inc., 1959.

(11) Hamming, R. W. Numerical Methods for Scientists and Engineers. New York: McGraw Hill Book Company, Inc., 1955.

(12) Hove, Larry E. "Synthesis of Controlled Processes by a Numerical Method." (unpublished Doctor's thesis, Oklahoma State University, 1964).

(13) Kumar, K. S. Prasanna, and R. Sridhar. "On the Identification of Control Systems by the Quasi-Linearization Method." IEEE Transactions on Automatic Control. Vol. AC-9. (April, 1964), 151-154.

(14) Lanczos, Cornelius. Applied Analysis. New Jersey: Prentice Hall, Inc., 1956.

(15) Matous, J. O., J. E. Bose, K. K. Gowdy, and M. A. Ross. "Hydraulic Control Systems." The Caterpillar Applied Control Series. Vol. 1. (unpublished document of the Fluid Power and Controls Laboratory, Oklahoma State University, 1965).

(16) McCollum, P. A., and J. L. Fikes. Fortran for the IBM 1620 and Selected Numerical Methods. Stillwater: Oklahoma State University, School of Electrical Engineering, 1962.

(17) Milne, W. E. Numerical Calculus. Princeton: Princeton University Press, 1949.

(18) Newton, G. C., L. A. Gould, and J. F. Kaiser. Analytical Design of Linear Feedback Controls. New York: John Wiley & Sons, Inc., 1957.

(19) Potts, T. F., G. N. Ornstein, and A. B. Clymer. "The Automatic Determination of Human and Other System Parameters." Proc. Western Joint Computer Conf. (May, 1961), 645-660.

(20) Shipley, D. D. "A Unified Approach to Synthesis of Linear Systems." IEEE Transactions on Automatic Control. Vol. AC-9. (April, 1964), 114-120.

(21) Truxal, John C. Automatic Feedback Control System Synthesis. New York: McGraw-Hill Book Company, Inc., 1955.

APPENDIX A

SOLUTION OF A SECOND-ORDER ORDINARY DIFFERENTIAL EQUATION WITH CONSTANT COEFFICIENTS BY THE RUNGE-KUTTA METHOD

APPENDIX A

SOLUTION OF A SECOND-ORDER ORDINARY DIFFERENTIAL EQUATION WITH CONSTANT COEFFICIENTS BY THE RUNGE-KUTTA METHOD

The Runge-Kutta method (16) is used to obtain an approximate numerical solution of an ordinary differential equation. Given N first-order equations, the following procedure indicates the steps adapted for application to a computer. Any equation of order M ($M > 1$) may be reduced to M first-order equations by appropriate substitution of variables.

Given: Y_i and X, initial values ($i = 1, 2, --, N$)

H = the interval $\qquad$ N = the number of equations

Solve for:

$$
\begin{array}{ll}
D_i(X,\ Y_i) & Y_i = Y_i + \tfrac{1}{2}HD_i \\
 & Q_i = HD_i \\
 & X = X + H/2 \\
D_i(X,\ Y_i) & A = 1 - \sqrt{0.5} \\
 & Y_i = Y_i + A(HD_i - Q_i) \\
 & Q_i = 2AHD_i + (1-3A)Q_i \\
D_i(X,\ Y_i) & A - 1 + \sqrt{0.5} \\
 & Y_i = Y_i + A(HD_i - Q_i) \\
 & Q_i = 2AHD_i + (1 - 3A)Q_i \\
 & X = X + H/2 \\
D_i(X,\ Y_i) & {}^{*}\ Y_i = Y_i + \dfrac{(HD_i - 2Q_i)}{6}
\end{array}
$$

*The Y_i's become the initial values for the next interval

If the equation to be solved is

$$\ddot{X} + B\dot{X} + CX + DX^3 + EX\dot{X} = 1$$

let $Y_1 = X$

$Y_2 = \dot{X}$

then the equations to be solved are

$$\dot{X} = Y_2$$

$$\ddot{X} = 1 - B\dot{X} - CX - DX^3 + EX\dot{X}.$$

The Fortran Program in this appendix will solve the specific equation

$$\ddot{X} + D(3)\dot{X} + D(4)X + D(5)X^3 + D(6)X\dot{X} = 1$$

by the Runge-Kutta method. By changing statement 1010, the program can be used to give a numerical solution to any second-order ordinary differential equation. A solution will be produced for each increment H of the independent variable. At every N^{th} calculation, the answer will be written out.

The given equation is to be solved for $\ddot{X}$, and this function is punched in fortran language into the statement card 1010. For use with statement 1010, the following equivalents exist: $D2 = \ddot{X}$; $Y2 = \dot{X}$; $Y1 = Y$.

The following data must either be entered into the program as typed statements or read from a data card:

N is the number of calculations between each write-out of answers.

H is the (X) increment between calculated points of answers.

XMAX is the maximum value of the independent variable.

Y1 is the initial value of the dependent variable.

Y2 is the intital value of the first derivative of the dependent variable.

Each variable is to be punched in a 10-column field. N must be right justified in the field of columns 1-10. H is to be punched in the field of columns 11-20; XMAX, 21-30; X, 31-40; Y1, 41-50; and Y2, 51-60.

```
      SUBROUTINE RK2(D)
C    FOURTH ORDER RUNGE KUTTA NONLINEAR PROGRAM
C
      DIMENSION D(52)
  502 FORMAT(1H0,12X,48HSOLUTION OF A SECOND-ORDER DIFFERENTIAL EQUATION
     1)
  503 FORMAT(23X,25HBY THE RUNGE-KUTTA METHOD)
  505 FORMAT(6X,23HINITIAL CONDITIONS ARE,/6X,4HT = ,F10.5)
  506 FORMAT(6X,11HINTERVAL = ,F10.5,/6X,12HMAXIMUM T = , F10.5)
  507 FORMAT(6X,28HANSWERS ARE PRINTED AT EVERY,I5,15H TH CALCULATION/)
  508 FORMAT(1H0,6X,3HNO.,6X,4HTIME,6X,4HDISP,7X,3HVEL,7X,3HACC,)
  509 FORMAT(6X,6HD(3) =,F10.5,/6X,6HD(4) =,F10.5,/6X,6HD(5) =,F10.5,
     1/6X,6HD(6) =,F10.5)
   98 FORMAT(I10,4F10.5)
    4 FORMAT(30X,11H*** END ***)
    2 FORMAT(I10,5F10.5)
      N=5
      H=0.02
      XMAX=10.0
      XX=0.0
      Y1=0.0
      Y2=0.0
      WRITE(6,502)
      WRITE(6,503)
      WRITE(6,505)XX
      WRITE(6,506)H,XMAX
      WRITE(6,507)N
      WRITE(6,509)D(3),D(4),D(5),D(6)
      WRITE(6,508)
      V=SQRT(0.5)
      L=1
      J=0
      K=0
      GO TO 1000
   10 Y1=Y1+0.5*H*D1
      Y2=Y2+0.5*H*D2
      Q1=H*D1
      Q2=H*D2
      XX=XX+H/2.
      GO TO 1000
   15 U=1.-V
      Y1=Y1+U*(H*D1-Q1)
      Y2=Y2+U*(H*D2-Q2)
      Q1=2.*U*H*D1+(1.-3.*U)*Q1
      Q2=2.*U*H*D2+(1.-3.*U)*Q2
      L=3
      GO TO 1000
   20 U=1.+V
      Y1=Y1+U*(H*D1-Q1)
      Y2=Y2+U*(H*D2-Q2)
      Q1=2.*U*H*D1+(1.-3.*U)*Q1
      Q2=2.*U*H*D2+(1.-3.*U)*Q2
      XX=XX+H/2.
      L=4
      GO TO 1000
   25 Y1=Y1+(H*D1-2.*Q1)/6.
      Y2=Y2+(H*D2-2.*Q2)/6.
      L=5
      GO TO 1000
   30 L=2
      IF (K) 35,40,35
   35 IF(K-N)50,40,40
   40 WRITE(6,98)J,XX,Y1,D1,D2
      IF(XX-XMAX) 55,45,45
   45 WRITE(6,4)
      GO TO 111
   50 K=K+1
      GO TO 10
   55 J=J+1
      K=1
      GO TO 10
 1000 D1=Y2
 1010 D2=D(3)*Y2+D(4)*Y1+D(5)*Y1*Y1*Y1+D(6)*Y1*Y2+1.0
      GO TO (30,15,20,25,30),L
  111 CONTINUE
      RETURN
      END
```

APPENDIX B

MATRIX ALGEBRA SUBROUTINES

APPENDIX B

MATRIX ALGEBRA SUBROUTINES

The matrix algebra subroutines (1) used in this investigation are described below. The matrix operations are written in single subscript notation to conserve core space within the computer. The Fortran listings describing the operations are also included for reference.

Fortran listings for the various matrix algebra subroutines are:

Subroutine Name	Description
WRTMAT (A)	Write matrix A.
MXM (A, B, C)	Postmultiply matrix A by matrix B. The product is matrix C.
INVERX (A, B)	Invert the matrix A and define $A^{-1} = B$.

The single subscript notation requires that the input format be as follows:

(1) The first element contains the number of rows, and the second element contains the number of columns of the matrix.

(2) For a column matrix B, the first element would be stored in B(3); the second, in B(4), etc.

(3) For square matrices, the elements are stored in memory by the following example Fortran statements:

```
     DO 60 I = 1, IROW        where IROW = number of rows
     DO 60 J = 1, JCOL              JCOL = number of columns
     II = (I-1)*JCOL + J + 2
  60 A(II) = C(I, J)
```

These statements place the ij^{th} element of matrix C in the correct memory location of matrix A.

Once the data is stored in this manner, the subroutines can be used.

APPENDIX B-I

FORTRAN PROGRAM FOR READING A MATRIX

```
$IBFTC RMAT
      SUBROUTINE RMAT(A)                                                RMAT001
      DIMENSION A(1)                                                    RMAT002
      COMMON KIN,KOUT                                                   RMAT003
    1 FORMAT(6X,I4,6X,I4)                                               RMAT004
    2 FORMAT(5E15.8)                                                    RMAT005
      READ (KIN,1) KA1,KA2                                              RMAT006
      IF(KA1.GT.0) GO TO 6                                              RMAT007
      WRITE(KOUT,200)                                                   RMAT008
  200 FORMAT(35H WE UNLOADED TAPES FROM MATRIX READ)                    RMAT009
      CALL EXIT                                                         RMAT010
    6 CONTINUE                                                          RMAT011
      KA1=A(1)                                                          RMAT012
      KA2=A(2)                                                          RMAT013
      L = A(1)                                                          RMAT014
      L1 = A(2)                                                         RMAT015
      J = L*L1 + 2                                                      RMAT016
      READ(KIN,2)(A(I),I=3,J)                                           RMAT017
      WRITE(KOUT,100)L,L1                                               RMAT018
  100 FORMAT(15H1THIS MATRIX IS,I4,3X,1HX,I4)                           RMAT019
      L2 = 3                                                            RMAT020
      DO 20 K = 1,L                                                     RMAT021
      L3 = L2 + L1 - 1                                                  RMAT022
      WRITE(KOUT,102)K                                                  RMAT023
  102 FORMAT(10X,5H ROW ,I4)                                            RMAT024
      WRITE(KOUT,101)(A(I),I=L2,L3)                                     RMAT025
  101 FORMAT(25X,6E15.6)                                                RMAT026
      L2 = L3 + 1                                                       RMAT027
   20 CONTINUE                                                          RMAT028
      RETURN                                                            RMAT029
      END                                                               RMAT030
```

APPENDIX B-II

FORTRAN PROGRAM FOR WRITING A MATRIX

```
$IBFTC WRTMAT  DECK
      SUBROUTINE WRTMAT(A)                                              WRMT001
      DIMENSION A(1)                                                    WRMT002
  100 FORMAT(15H1THIS MATRIX IS,I4,3X,1HX,I4)                          WRMT003
  101 FORMAT(20X,1P6E16.7)                                              WRMT004
  102 FORMAT(10X,5H ROW ,I4)                                            WRMT005
      COMMON KIN,KOUT                                                   WRMT006
      L = A(1)                                                          WRMT007
      L1 = A(2)                                                         WRMT008
      L2 = 3                                                            WRMT009
      J = L*L1 + 2                                                      WRMT010
      WRITE(KOUT,100)L,L1                                               WRMT011
      DO 20 K = 1,L                                                     WRMT012
      L3 = L2 + L1 - 1                                                  WRMT013
      WRITE(KOUT,102)K                                                  WRMT014
      WRITE(KOUT,101) (A(I),I=L2,L3)                                    WRMT015
      L2 = L3 + 1                                                       WRMT016
   20 CONTINUE                                                          WRMT017
      RETURN                                                            WRMT018
      END                                                               WRMT019
```

APPENDIX B-III

FORTRAN PROGRAM FOR MULTIPLYING TWO MATRICES

```
$IBFTC MXM      DECK
      SUBROUTINE MXM(A,B,C)
      DIMENSION A(1),B(1),C(1)
  100 FORMAT(1H0,49HTHE MATRICES ARE NOT CONFORMAL FOR MULTIPLICATION,2(
     15X,I4,2HX ,I4))
      COMMON KIN,KOUT
      IROWA=A(1)
      ICOLA=A(2)
      IROWB=B(1)
      ICOLB=B(2)
      IF(ICOLA-IROWB.EQ.0) GO TO 4
      WRITE(KOUT,100) IROWA,ICOLA,IROWB,ICOLB
      GO TO 6
4     N=IROWA*ICOLB+2
      DO 5 I=1,N
5     C(I)=0.0
      IX=3
      I=3
      J=3
      K=3
      KX=3
      DO 10 M=1,IROWA
      DO 9 N=1,ICOLB
      DO 8 NX=1,ICOLA
      C(J)=C(J)+A(I)*B(K)
      I=I+1
8     K=K+ICOLB
      I=IX
      J=J+1
      KX=KX+1
9     K=KX
      IX=IX+ICOLA
      I=IX
      K=3
10    KX=3
    6 C(1)=A(1)
      C(2)=B(2)
      RETURN
      END
```

APPENDIX B-IV

FORTRAN PROGRAM FOR INVERTING A MATRIX

```
$IBFTC INVERX
      SUBROUTINE INVERX(A,B)
      DIMENSION A(1),B(1)
      DET = 1.0
      N = A(1)
      L10 = N**2 + 2
      DO 1 I = 1,L10
    1 B(I) = 0.
      B(1) = N
      B(2) = N
      L9 = N + 1
      DO 2 I = 3,L10,L9
    2 B(I) = 1.0
      JK = N - 1
      J = 3
      N1 = 3
      N2 = N + 2
      J0 = N - 1
      J2 = N + 3
      J4 = 3
      DO 300 L1 = 1,JK
      NR = (J + N - 2)/(N + 1)
      NR1 = NR
      NRI = N - NR
      JN1 = J + N
      IF(NRI.LT.1) GO TO 900
      IF(NRI.GT.1) GO TO 804
  800 AMAX=ABS(A(J))
      AMXA=ABS(A(JN1))
      IF(AMAX.GE.AMXA) GO TO 900
  801 N5 = J - NR + 1
      N6 = N5 + N - 1
      IAD = N
  802 DO 803 IT = N5,N6
      IT6 = IT + IAD
      ATEM = A(IT)
      A(IT) = A(IT6)
      A(IT6) = ATEM
      ATEM = B(IT)
      B(IT) = B(IT6)
  803 B(IT6) = ATEM
      GO TO 900
  804 J11 = J + N + 1
      J10 = J + N
      AMAX=ABS(A(J))
      DO 807 IT = 1,NRI
      AMXA=ABS(A(J10))
      IF(AMAX.GE.AMXA) GO TO 806
  805 AMAX = AMXA
      NR1 = (J11 + N - 2)/(N + 1)
  806 J10 = J10 + N
  807 J11 = J11 + N + 1
      N5 = J - NR + 1
      N6 = N5 + N - 1
      ITEM = NR1 - NR
      IAD = ITEM*N
      IF(IAD.GT.0) GO TO 802
  900 CONTINUE
      DENOM = A(J)
      IF(DENOM.EQ.0.0) GO TO 51
   50 IF(IAD.GT.0) GO TO 701
  700 DET = DET*DENOM
```

```
      GO TO 702
  701 DET = DET*(-DENOM)
  702 DO 100 J1 = N1,N2
      A(J1) = A(J1)/DENOM
  100 B(J1) = B(J1)/DENOM
      J3 = J4
      N3 = N2 + 1
      N4 = N2 + N
      DO 200 L = 1,JO
      AMULT = A(J2)
      DO 101 J1 = N3,N4
      A(J1) = A(J1) - AMULT*A(J3)
      B(J1) = B(J1) - AMULT*B(J3)
  101 J3 = J3 + 1
      J2 = J2 + N
      J3 = J4
      N3 = N3 + N
  200 N4 = N4 + N
      N1 = N1 + N
      N2 = N2 + N
      JO = JO - 1
      J = J + N + 1
      J2 = J + N
  300 J4 = J4 + N
      DENOM = A(J)
      IF(DENOM.EQ.0.0) GO TO 51
   60 A(J) = A(J)/DENOM
      DET = DET*DENOM
      LT = J - N + 1
      DO 400 J1 = LT,J
  400 B(J1) = B(J1)/DENOM
      JO = JK
      J2 = J - N
      J4 = J - N + 1
      N2 = J2 - N
      DO 600 L1 = 1,JK
      J3 = J4
      N3 = N2 + 1
      N4 = N2 + N
      DO 500 L = 1,JO
      AMULT = A(J2)
      DO 401 J1 = N3,N4
      A(J1) = A(J1) - AMULT*A(J3)
      B(J1) = B(J1) - AMULT*B(J3)
  401 J3 = J3 + 1
      J3 = J4
      J2 = J2 - N
      N3 = N3 - N
  500 N4 = N4 - N
      N2 = N2 - N
      JO = JO - 1
      J = J - N - 1
      J2 = J - N
  600 J4 = J4 - N
      IE = 1
  703 RETURN
   51 IE = 0
      GO TO 703
      END
```

APPENDIX C

FORTRAN PROGRAM TO OBTAIN ITERATIVE SOLUTION USING

STRAIGHT-LINE APPROXIMATION OF THE

STATE VARIABLES

APPENDIX C

FORTRAN PROGRAM TO OBTAIN ITERATIVE SOLUTION USING STRAIGHT-LINE APPROXIMATION OF THE STATE VARIABLES

The Fortran Program presented in this appendix may be broken down into the following major divisions:

(1) Generation of the Straight-Line Approximation

(2) Generation of the Elements of the Normal Equations

(3) Solving for the Unknown Coefficients A_{iq} of Equation (3-21)

(4) Solving the Nonlinear Differential Equation Using the Coefficients Obtained by the Least Squares Method

(5) Checking and Adjusting of the Output Response

The major divisions of the program are independent and will be discussed separately.

Generation of the Straight-Line Approximation

The input data which is required by the program is used only in the generation of the straight-line approximation. Three input values are read into memory. They are:

TRT - rise time

TST - settling time

OS - overshoot

For example, for a system with the following desired characteristics

$T_{RT} = 5.0$

$T_{ST} = 6.0$

$OS = 10\%$

the values of the three input variables would be

TRT = 5.0

TST = 6.0

OS = 1.10.

These values are punched in fields of 10 columns. This data is then used to generate the straight-line approximation. The approximation is generated by Fortran statements 707 through 7879. The displacement is stored in the array X(I), the velocity in array X1(I), and the acceleration in array X2(I).

Generation of the Elements of the Normal Equations

The variable IROW is the number of rows and JCOL is the number of columns of the matrix U (see Equation 3-21). The elements of U are generated and stored in double subscript notation. For example, U(1,1) is stored in array CO(1,1). The elements are generated and stored in memory in statements 51 through 61.

The elements of Y are stored consecutively in array B(I) with Y_{i1} in B(3), Y_{i2} in B(4), etc. B(1) contains the number of columns and B(2) the number of rows. Generating and storing these elements in this way make the data accessible by the matrix subroutines in Appendix B. The elements of Y are generated and stored in array B by statements 60 through 600.

Solving for the Unknown Coefficients A_{iq} of Equation (3.21)

Once the elements of the normal equations have been generated, the coefficients A_{iq} are obtained by calling the following two subroutines:

(1) INVERX(A,AI,DET,IE)

(2) MXM(AI,B,D)

The first subroutine takes the inverse of A and stores it in matrix AI. The second subroutine multiplies AI by the matrix B and stores the value of the coefficients A_{iq} in matrix D.

Solving the Nonlinear Differential Equations Using the Coefficients Obtained by the Least Squares Method

The required input data necessary for the Runge-Kutta solution is defined in Appendix A. The displacement, velocity, and acceleration obtained from the Runge-Kutta solution are stored in arrays RD, RV, and RA, respectively. This operation is defined by statements 40 through 45. The maximum value of the displacement is stored in DMAX.

Checking and Adjusting of the Output Response

The checking of the output response is made in statements 45 up to but not including statement 50. If the rise time fails to meet the desired specification, the straight-line parameter T_1 is adjusted and the program is transferred back to statement 717 by statement 177 up to statement 178. If the desired response meets all desired characteristics, the program is transferred out of the loop by the statements GO TO 707.

Adding of Additional Terms

Additional terms may be added to the program by making the following changes:

(1) Make IROW and JCOL equal to the number of terms used.

(2) Add additional statement card to DO loop 51.

(3) Change statement 1010 to include the additional change.

For example, assume that the term $X\dot{X}^2$ is to be added to the program. Set the following values into the program:

(2) IROW = JCOL = 5

(2) Add the card <u>DD(I,5) = DD(I,4) * X1(I)</u> to the DO loop 51

(3) Change statement 1010 to: 1010 D2 = D(3)*Y2+D(4)*Y1+D(5)*Y1*Y1*Y1+D(6)*Y1*Y2+D(7)*Y1*Y2*Y2+1.0

General Comments

The value of the delay time T_d entered into the Fortran program as DELY was set equal to 0.1 for rise time values in the range of 2 seconds to 4 seconds. The value of DELY was equal to 0.8 for a rise time equal to 1 second. The iterative procedure does not include the necessary steps to determine this value. DELY was initially equal to 0.1 and increased in increments of 0.1 until a "best" value was obtained. This value was then substituted into the Fortran program as a constant.

The number of data points (M in the Fortran program) used in the method was found by dividing the maximum time of the straight-line procedure, TMAX, by the time increment, DELT.

The output listing of the digital program contains the following data printed from right to left:

(1) Number of points taken in the procedure.
(2) Time of the continuous solution.
(3) Straight-line displacement X(I).
(4) Continuous displacement.
(5) Straight-line velocity X1(I).
(6) Continuous velocity.
(7) Straight-line acceleration X2(I).
(8) Continuous acceleration.

If the product NH is made equal to DELT, then the straight-line data will correspond to the continuous solution time.

If the settling time T_{ST} failed to meet specifications, the value of T_{ST} was reduced. The minimum value of the settling time is a function of the specified overshoot and equation form. If a desired minimum settling time is not found in the iterative procedure, a reduction in the specified overshoot must be made.

```
$JOB    WATFOR      BOSE                                  2527-40048
BFTC ST LINE
$IBJOB          GO,MAP,NODECK
      REAL M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,MA,MB,MC
      DIMENSION A(227),AI(227),B(52),D(52)
      DIMENSION RD(200),RV(200),RA(200)
      DIMENSION DD(250,10),C(10,10)
      DIMENSION X(200),X1(200),X2(200)
      DIMENSION CO(15,15)
      EQUIVALENCE (DD(1,1),X1(1)),(DD(1,2),X(1)),(DD(1,5),X2(1))
      COMMON  KIN, KOUT
      KIN = 5
      KOUT = 6
  999 FORMAT(8F10.5)
   27 FORMAT(6X,3HWNT,6X,4HZETA,6X,4HDELT,6X,4HTMAX)
   21 FORMAT(2F10.5)
   22 FORMAT(3F10.5)
   11 FORMAT(2F10.5)
  939 FORMAT(I2)
  333 FORMAT(I10,7F10.5)
  223 FORMAT(8I10)
   26 FORMAT(4F10.5)
   24 FORMAT(1H0,6X,26HPERFORMANCE SPECIFICATIONS)
    3 FORMAT(I10,5F10.5)
   77 FORMAT(8F10.5)
    4 FORMAT(30X,11H*** END ***)
    2 FORMAT(I10,5F10.5)
   23 FORMAT(1H0,7X,3HNO.,6X,4HTIME,6X,4HX(T),15X,5HX1(T),15X,5HX2(T))
  101 FORMAT(F10.5)
  929 FORMAT(6X,6HDELT =,F10.5/
     1       6X,6HTMAX =,F10.5/
     2       6X,11HRISE TIME =,F10.5/
     3       6X,11HOVERSHOOT =,F10.5/
     4       6X,15HSETTLING TIME =,F10.5/
     5       6X,22HINITIAL ACCELERATION =,F10.5)
 1001 FORMAT(1H0,25X,24HTHE MAXIMUM OVERSHOOT IS,F10.5,2X,7HPERCENT)
 1002 FORMAT(1H0,25X,48HRISE TIME FAILS TO MEET DESIRED SPECIFICATION OF
     1,F10.5,2X,7HSECONDS)
 1003 FORMAT(1H0,25X,48HRISE TIME *** MEETS *** DESIRED SPECIFICATION OF
     1,F10.5,2X,7HSECONDS)
C
C     RISE TIME (TRT), SETTLING TIME (TST), AND OVERSHOOT (OS) MUST
C     BE READ INTO MEMORY IN A 10 COLUMN FIELD
C
C     STATEMENTS 707 THROUGH 786 GENERATE THE STRAIGHT LINE APPROXIMATION
C
C     DELT AND TMAX ARE THE INCREMENT AND MAXIMUM VALUES OF THE
C     STRAIGHT LINE APPROXIMATION
C
C     X(I)=DISPLACEMENT   X1(I)=VELOCITY   X2(I)=  ACCELERATION
C
C     ADD=CORRECTED VELOCITY CO    ACC=INITIAL ACCELERATION
C
      WRITE(6,24)
      DELY=0.1
      READ(5,21) DELT,TMAX
      CA1=0.50
  707 CONTINUE
      READ(5,22)TRT,TST,OS
      TOS=TRT+0.5*(TST-TRT)
      C6=1.0
      C7=1.0
```

```
      RT=1.0
      T1=TRT*C6
  717 CONTINUE
      DMAX=0.0
      ST=1.0
      T2=TOS*C7
      BA=1.0-DELY
      T3=TST*C7
      T4=DELY*TRT
      T10=T4
      M1=RT/T1
      M2=(OS-RT)/(T2-T1)
      M3=(ST-OS)/(T3-T2)
      ADD=1.0/(T4*0.5+T1-T4)
      M4=ADD/T4
      ACC=ADD/(0.5*T4)
      M8=-ACC/T10
      B1=0.0
      B2=(T2*RT-T1*OS)/(T2-T1)
      B3=(T3*OS-T2)/(T3-T2)
      B4=0.0
      B8=ACC
      I1=IFIX(T1/DELT)
      I2=IFIX(T2/DELT)
      I3=IFIX(T3/DELT)
      I4=IFIX(T4/DELT)
      I10=IFIX(T10/DELT)
      F=TMAX/DELT+1.
      M=F
      DO 7077 J=1,10
      DO 7077 I=1,250
      DD(I,J)=0.0
 7077 CONTINUE
      DO 777 I=1,200
      X(I)=0.0
      X1(I)=0.0
      X2(I)=0.0
  777 CONTINUE
      DO 778 I=1,I1
      E=I-1
      T=E*DELT
      X(I)=M1*T+B1
  778 CONTINUE
      DO 779 I=I1,I2
  779 X(I)=M2*FLOAT(I)*DELT+B2
      DO 780 I=I2,I3
  780 X(I)=M3*FLOAT(I)*DELT+B3
      DO 781 I=I3,M
  781 X(I)=1.0
      DO 782 I=1,I4
      E=I-1
      T=E*DELT
  782 X1(I)=M4*T+B4
      DO 7831 I=I4,I1
 7831 X1(I)=ADD
      DO 7832 I=I1,I2
 7832 X1(I)=M2
      DO 7834 I=I2,I3
 7834 X1(I)=M3
      DO 7835 I=I3,M
 7835 X1(I)=0.0
      I10=I10+3
```

```
      DO 786 I=1,I10
      E=I-1
      T=E*DELT
      X2(I)=M8*T+B8
      IF(X2(I).LT.0.0) X2(I)=0.0
  786 CONTINUE
      IROW=2
      JCOL=2
      TA=CA1*(T2-T1)
      TB=CA1*(T3-T2)
      TA1=T1-TA
      TA2=T1+TA
      TA3=T2-TB
      TA4=T2+TB
      TA5=T3-TB
      IA1=IFIX(TA1/DELT)
      IA2=IFIX(TA2/DELT)
      IA3=IFIX(TA3/DELT)
      IA4=IFIX(TA4/DELT)
      IA5=IFIX(TA5/DELT)
      AA=(ADD-M2)/(TA1-TA2)
      BB=(M2-M3)/(TA3-TA4)
      CC=-M3/(T3-TA5)
      DO  7877 I=IA1,IA2
 7877 X2(I)=AA
      DO 7878 I=IA3,IA4
 7878 X2(I)=BB
      DO 7879 I=IA5,I3
 7879 X2(I)=CC
C
C     NORMAL EQUATIONS MUST BE GENERATED NEXT
```

```
      JBCOL=1
      IROW=4
      JCOL=4
      DO 51 I=1,M
      DD(I,3)=X(I)**3
      DD(I,4)=X1(I)*X(I)
C     IF ADDITIONAL TERMS ARE ADDED, THE ADDED TERM MUST BE GENERATED
C     IN THIS LOOP
   51 CONTINUE
      WRITE(6,4)
      WRITE(6,4)
      DO 61 I=1,IROW
      DO 61 J=1,JCOL
      SUM=0.0
      DO 56 K=1,M
   56 SUM=SUM+DD(K,I)*DD(K,J)
      CO(I,J)=SUM
   61 CO(J,I)=SUM
   65 CONTINUE
      WRITE(6,4)
C     SQUARE MATRIX
C     CONVER TO A(I) FOR MATRIX PACKAGE
      A(1) = IROW
      A (2) = JCOL
      DO 60 I=1,IROW
      DO 60 J=1,JCOL
      II=(I-1)*JCOL+J+2
      A (II) = CO (I,J)
   60 CONTINUE
      WRITE(6,4)
      DO 600 J=1,4
      SUM=0.
      DO 550 K=1,M
  550 SUM=SUM+DD(K,5)*DD(K,J)-DD(K,J)
C         DD(K,J)  HAS BEEN SUBTRACTED FOR THE STEP INPUT TO BE VALID
      B(1)=IROW
      B(2)=JBCOL
      KK=J+2
      B(KK)=SUM
  600 CONTINUE
      CALL INVERX (A, AI, DET, IE)
      CALL MXM(AI, B, D)
      CALL WRTMAT (D)
C    FOURTH ORDER RUNGE KUTTA NONLINEAR PROGRAM
      WRITE(6,24)
      WRITE(6,929) DELT,TMAX,TRT,OS,TST,ACC
      WRITE(6,23)
      P=1.0
      H=0.02
      N=5
      XMAX=8.0
      XX=0.0
      Y1=0.0
      Y2=0.0
      V=SQRT(0.5)
      L=1
      J=0
      K=0
      GO TO 1000
   10 Y1=Y1+0.5*H*D1
      Y2=Y2+0.5*H*D2
      Q1=H*D1
```

```
      Q2=H*D2
      XX=XX+H/2.
      GO TO 1000
   15 U=1.-V
      Y1=Y1+U*(H*D1-Q1)
      Y2=Y2+U*(H*D2-Q2)
      Q1=2.*U*H*D1+(1.-3.*U)*Q1
      Q2=2.*U*H*D2+(1.-3.*U)*Q2
      L=3
      GO TO 1000
   20 U=1.+V
      Y1=Y1+U*(H*D1-Q1)
      Y2=Y2+U*(H*D2-Q2)
      Q1=2.*U*H*D1+(1.-3.*U)*Q1
      Q2=2.*U*H*D2+(1.-3.*U)*Q2
      XX=XX+H/2.
      L=4
      GO TO 1000
   25 Y1=Y1+(H*D1-2.*Q1)/6.
      Y2=Y2+(H*D2-2.*Q2)/6.
      L=5
      GO TO 1000
   30 L=2
      IF (K) 35,40,35
   35 IF(K-N)50,40,40
   40 JJ=J+1
      RD(JJ)=Y1
      RV(JJ)=D1
      RA(JJ)=D2
      WRITE(6,333)J,XX,X(JJ),Y1,X1(JJ),D1,X2(JJ),D2
      IF(RD(JJ).GT.DMAX)   DMAX=RD(JJ)
      TEST=(RD(JJ)-1.0)*P
      IF(TEST)13,13,12
   12 TT=XX
      WRITE(6,101)TT
      P=0.0
   13 CONTINUE
      IF(XX-XMAX) 55,45,45
C
   45 CONTINUE
      BETA=OS-1.0
      AL1=OS-BETA*0.10
      AL2=OS+BETA*0.10
      PERC=(DMAX-1.0)*100.0
      IF(PERC.LT.0.0)PERC=0.0
      WRITE(6,1001)PERC
      IKK=IFIX(TRT/DELT)+1
      IKS=IFIX(TST/DELT)+1
      IF(RD(IKK)-1.0)177,178,179
  177 CONTINUE
      T1=T1+(TRT-TT)*2.0
      WRITE(6,1002)TRT
      WRITE(6,999) C7,T1,DMAX,TRT,OS,CA1,   RD(IKS),RD(IKK)
      GO TO 717
  179 CONTINUE
      IF(TT.LT.0.9*TRT) GO TO 971
      GO TO 178
  971 T1=T1+(TRT-TT)*1.5
      GO TO 717
  178 CONTINUE
      IF(DMAX-OS)1177,1178,1177
 1178 CONTINUE
```

```
      IF(RD(IKS).GT.1.05)GO TO 3111
      WRITE(6,999) C7,T1,DMAX,TRT,OS,CA1,  RD(IKS),RD(IKK)
      GO TO 707
3111  CONTINUE
      CA1=CA1-0.1
      WRITE(6,999) C7,T1,DMAX,TRT,OS,CA1,  RD(IKS),RD(IKK)
      GO TO 717
1177  CONTINUE
      IF(DMAX.GT.AL1) GO TO 3711
      C7=C7-(DMAX-OS)*5.0
      WRITE(6,999) C7,T1,DMAX,TRT,OS,CA1,  RD(IKS),RD(IKK)
      GO TO 717
3711  CONTINUE
      IF(DMAX.LT.AL2) GO TO 3311
      C7=C7-(DMAX-OS)*3.0
      WRITE(6,999) C7,T1,DMAX,TRT,OS,CA1,  RD(IKS),RD(IKK)
      GO TO 717
3311  IF(RD(IKS).LT.1.05) GO TO 707
      WRITE(6,999) C7,T1,DMAX,TRT,OS,CA1,  RD(IKS),RD(IKK)
      GO TO 3111
  50  K=K+1
      GO TO 10
  55  J=J+1
      K=1
      GO TO 10
1000  D1=Y2
1010  D2=D(3)*Y2+D(4)*Y1+D(5)*Y1*Y1*Y1+D(6)*Y1*Y2+1.
      GO TO (30,15,20,25,30),L
      END
```

APPENDIX D

FORTRAN PROGRAM FOR OBTAINING ITERATIVE SOLUTION USING STRAIGHT-LINE APPROXIMATION OF THE STATE VARIABLES. SPECIAL CASE: TWO FIXED COEFFICIENTS.

APPENDIX D

FORTRAN PROGRAM FOR OBTAINING ITERATIVE SOLUTION USING STRAIGHT-LINE APPROXIMATION OF THE STATE VARIABLES. SPECIAL CASE: TWO FIXED COEFFICIENTS.

The Fortran program presented is written to solve the following system equation:

$$\ddot{X} + 0.36\dot{X} + 0.24X + A_3X^3 + A_4X\dot{X} = 1.$$

The coefficients of the linear terms are fixed and the coefficients A_3 and A_4 are to be determined in the analysis method for a specific system response.

The coefficients of the linear terms are entered into the program following statement 51 where D_3 is equal to -0.36 and D_4 is equal to -0.24. The sign of the coefficients are negative to agree with the state variable representation. The Fortran listing of this appendix must be preceded by the straight-line program of Appendix C. For the following input data

$$T_{RT} = 2.0$$
$$T_{ST} = 3.0$$
$$OS = 10\%$$

the value of the coefficients A_3 and A_4 are equal to 0.801 and 1.4039, respectively. The value of the delay time, DELY, was equal to 0.1.

```
      DO 51 I=1,M
      DD(I,3)=X(I)**3
      DD(I,4)=X1(I)*X(I)
   51 CONTINUE
      D3=-0.36
      D4=-0.24
      SUM=0.0
      DO 770 I=1,M
      SUM=SUM+DD(I,4)*DD(I,4)
  770 CO(2,2)=SUM
      SUM=0.0
      DO 771 I=1,M
      SUM=SUM+DD(I,3)*DD(I,3)
  771 CO(1,1)=SUM
      SUM=0.0
      DO 772 I=1,M
      SUM=SUM+DD(I,3)*DD(I,4)
      CO(2,1)=SUM
  772 CO(1,2)=SUM
      IL=2
      A(1)=IL
      A(2)=IL
      DO 773 I=1,IL
      DO 773 J=1,IL
      II=(I-1)*IL+J+2
      A(II)=CO(I,J)
  773 CONTINUE
      B(1)=2
      B(2)=1
      SUM=0.0
      DO 774 I=1,M
  774 SUM=SUM+DD(I,3)*(DD(I,5)-D3*DD(I,1)-D4*DD(I,2)-1.)
      B(3)=SUM
      SUM=0.0
      DO 775 I=1,M
  775 SUM=SUM+DD(I,4)*(DD(I,5)-D3*DD(I,1)-D4*DD(I,2)-1.)
      B(4)=SUM
      CALL WRTMAT(A)
      CALL INVERX(A,AI,DET,IE)
      CALL WRTMAT(AI)
      CALL WRTMAT(B)
      CALL MXM(AI,B,D)
      CALL WRTMAT(D)
      D(5)=D(3)
      D(6)=D(4)
      D(3)=D3
      D(4)=D4
      WRITE(6,26) D(3),D(4),D(5),D(6)
      WRITE(6,929) DELT,TMAX,TRT,OS,TST,ACC
      WRITE(6,23)
      P=1.0
      H=0.01
      N=5
      XMAX=5.0
      XX=0.0
      Y1=0.0
      Y2=0.0
      V=SQRT(0.5)
      L=1
      J=0
      K=0
      GO TO 1000
```

```
 10 Y1=Y1+0.5*H*D1
    Y2=Y2+0.5*H*D2
    Q1=H*D1
    Q2=H*D2
    XX=XX+H/2.
    GO TO 1000
 15 U=1.-V
    Y1=Y1+U*(H*D1-Q1)
    Y2=Y2+U*(H*D2-Q2)
    Q1=2.*U*H*D1+(1.-3.*U)*Q1
    Q2=2.*U*H*D2+(1.-3.*U)*Q2
    L=3
    GO TO 1000
 20 U=1.+V
    Y1=Y1+U*(H*D1-Q1)
    Y2=Y2+U*(H*D2-Q2)
    Q1=2.*U*H*D1+(1.-3.*U)*Q1
    Q2=2.*U*H*D2+(1.-3.*U)*Q2
    XX=XX+H/2.
    L=4
    GO TO 1000
 25 Y1=Y1+(H*D1-2.*Q1)/6.
    Y2=Y2+(H*D2-2.*Q2)/6.
    L=5
    GO TO 1000
 30 L=2
    IF (K) 35,40,35
 35 IF(K-N)50,40,40
 40 JJ=J+1
    RD(JJ)=Y1
    RV(JJ)=D1
    RA(JJ)=D2
    WRITE(6,333)J,XX,X(JJ),Y1,X1(JJ),D1,X2(JJ),D2
    IF(RD(JJ).GT.DMAX)  DMAX=RD(JJ)
    TEST=(RD(JJ)-1.0)*P
    IF(TEST)13,13,12
 12 TT=XX
    WRITE(6,101)TT
    P=0.0
 13 CONTINUE
    IF(XX-XMAX) 55,45,45
 45 CONTINUE
    BETA=OS-1.0
    AL1=OS-BETA*0.10
    AL2=OS+BETA*0.10
    PERC=(DMAX-1.0)*100.0
    IF(PERC.LT.0.0)PERC=0.0
    WRITE(6,1001)PERC
    IKK=IFIX(TRT/DELT)+1
    IKS=IFIX(TST/DELT)+1
    IF(RD(IKK)-1.0)177,178,179
177 CONTINUE
    T1=T1+(TRT-TT)*2.0
    WRITE(6,1002)TRT
    GO TO 717
179 CONTINUE
    IF(TT.LT.0.9*TRT) GO TO 971
    GO TO 178
971 T1=T1+(TRT-TT)*1.5
    GO TO 717
178 CONTINUE
    IF(DMAX-OS)1177,1178,1177
```

```
1178 CONTINUE
     IF(RD(IKS).GT.1.05)GO TO 3111
     GO TO 707
3111 CONTINUE
     CA1=CA1-0.1
     GO TO 717
1177 CONTINUE
     IF(DMAX.GT.AL1) GO TO 3711
     C7=C7-(DMAX-OS)*3.0
     GO TO 717
3711 CONTINUE
     IF(DMAX.LT.AL2) GO TO 3311
     C7=C7-(DMAX-OS)*3.0
     GO TO 717
3311 IF(RD(IKS).LT.1.05) GO TO 707
     GO TO 3111
  50 K=K+1
     GO TO 10
  55 J=J+1
     K=1
     GO TO 10
1000 D1=Y2
1010 D2=D(3)*Y2+D(4)*Y1+D(5)*Y1*Y1*Y1+D(6)*Y1*Y2+1.
     GO TO (30,15,20,25,30),L
     END
```

VITA

James Edwin Bose

Candidate for the Degree of

Doctor of Philosophy

Thesis: DESIGN OF NONLINEAR SECOND-ORDER DYNAMIC SYSTEMS TO MEET ARBITRARY PERFORMANCE CRITERIA

Major Field: Engineering

Biographical

Personal Data: Born August 17, 1937, in Bessie, Oklahoma, the son of Edward and Margaret Bose.

Education: Graduated from Clinton High School, Clinton, Oklahoma, in May, 1955; received the degree of Bachelor of Science in Mechanical Engineering from Oklahoma State University, May, 1960; received the degree of Master of Science in Mechanical Engineering from Oklahoma State University, August, 1961; completed the requirements for the degree of Doctor of Philosophy, July, 1966.

Professional Experience: Employed at Oklahoma State University from September, 1960, to September, 1961, as a graduate assistant; from September, 1961, to September, 1963, as a graduate research assistant; and from September, 1963, to September, 1966, as an instructor.